(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

Simulated Annealing in Sonar Track Detection

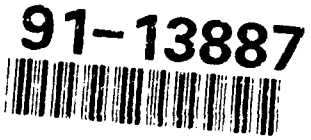
by

Tung-Sheng Chen

December 1990

Thesis Advisor: Chin-Hwa Lee

Approved for public release; distribution is unlimited.



Unclassified

Security Classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassifie.	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report: Approved for public release; distribution is unlimited.
2b Declassification/Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization: Naval Postgraduate School; 6b Office Symbol: 62	7a Name of Monitoring Organization: Naval Postgraduate School
6c Address (city, state, and ZIP code): Monterey, CA 93943-5000	7b Address (city, state, and ZIP code): Monterey, CA 93943-5000
8a Name of Funding/Sponsoring Organization; 8b Office Symbol (If Applicable)	9 Procurement Instrument Identification Number
8c Address (city, state, and ZIP code)	10 Source of Funding Numbers: Program Element Number; Project No; Task No; Work Unit Accession No

11 Title (Include Security Classification)
SIMULATED ANNEALING IN SONAR TRACK DETECTION

12 Personal Author(s) Tung-Sheng Chen

13a Type of Report	13b Time Covered	14 Date of Report (year, month,day)	15 Page Count
Master's Thesis	From To	1990 December	71

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (continue on reverse if necessary and identify by block number)
Field	Group	Subgroup	Simulated Annealing, Sonar Track Detection, Combinatorial optimization

19 Abstract (continue on reverse if necessary and identify by block number

Classical edge detectors cannot handle the problem of noisy sonar track detection because they are too sensitive to noise. This thesis uses Simulated Annealing for the solution by formulating the sonar track detection problem as an optimization problem. The experimental results are much better than those of classical edge detectors. Fairly good results can be obtained on data with SNR down to -18 dB where most classical edge detectors fail.

20 Distribution/Availability of Abstract: [X] unclassified/unlimited [ ] same as report [ ] DTIC users	21 Abstract Security Classification: Unclassified
22a Name of Responsible Individual: Chin-Hwa Lee	22b Telephone (Include Area code): (408) 655-0242; 22c Office Symbol: 62Le

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted

All other editions are obsolete

security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

Simulated Annealing in Sonar Track Detection

by

Tung-Sheng Chen
Captain, Republic of China Army
B.S., CHUNG CHENG INSTITUTE OF TECHNOLOGY, 1986

Submitted in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1990

Author: Tung-Sheng Chen
Tung-Sheng Chen

Approved by: Chin-Hwa Lee
Chin-Hwa Lee, Thesis Advisor

Charles W. Therrien
Charles W. Therrien, Second Reader

Michael A. Morgan
Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

Classical edge detectors cannot handle the problem of noisy sonar track detection because they are too sensitive to noise. This thesis uses Simulated Annealing for the solution by formulating the sonar track detection problem as an optimization problem. The experimental results are much better than those of classical edge detectors. Fairly good results can be obtained on data with SNR down to -18 dB where most classical edge detectors fail.

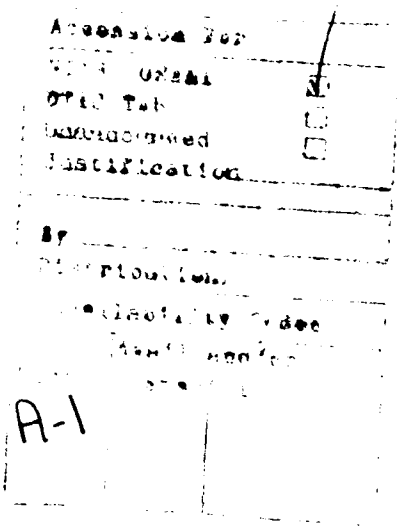


TABLE OF CONTENTS

LIST OF FIUGRES

ACKNOWLEDGEMENT

I wish to express my gratitude to my thesis adviser, Dr. Chin-Hwa Lee, in his advice and assistance in the completion of this thesis. I would also like to thank Dr. Charles W. Therrien who served as my second reader. At last, I would like to thank my wife Min-Hui Wu for giving me the love and care that supported me to finish this thesis.

I. INTRODUCTION

This thesis is an investigation of applying the **Simulated Annealing(SA)** [Ref. 1] algorithm to the problem of sonar track detection. The acoustic track recognition is difficult because of the presence of large quantities of noise from all sources.

Classical edge detectors are not robust enough to deal with noisy tracks, and a high degree of false and fragmented edges will be produced. Simulated Annealing may be used to solve the problem by formulating the sonar track detection problem as an optimization problem. Since Simulated Annealing is very effective in finding the global optimal solution, track detection using Simulated Annealing has very good potential.

T. Hibbert has written a report of "Sonar Track Detection Using Simulated Annealing" [Ref. 2]. In this report, multi-track data is used. Tracks at low frequency are detected with good results, while the tracks at high frequency are not detected due to the noise. Further, the tracks produced in this report are fragmented.

Essentially, the optimization method of Simulated Annealing used in this thesis is concentrated on local track fine tuning. The use of local track tuning methods studied here restricts the application of this method to a small

region of the lofargram. Therefore some future improvement is needed. The test image data used here includes single track data with various SNR values, multi-track data, and sweep track data. The experimental results show that the Simulated Annealing detection algorithm can be very sensitive to a single track with very low SNR. Most classical edge detectors do not work at all with these images. The Simulated Annealing algorithm also works with the multi-track and sweep track images which are shown in the later chapters.

A brief background of the theory of Simulated Annealing is given in Chapter II. Detailed treatments of the experimental procedure and results are presented in Chapters III and IV. In Chapter III, three cases of single track detection problems are described. In Chapter IV, the cases of multi-track and sweep track detection problems are reported. Finally, in the last chapter conclusions and some unresolved questions are discussed.

II THEORETICAL DISCUSSION

This chapter provides a brief background of the Simulated Annealing algorithm. Some basic terminologies are introduced. The procedures of Simulated Annealing, consisting of four steps, are also discussed.

A. Acoustic track detection and Simulated Annealing

Simulated Annealing gets its name from an analogy with the physical process of annealing. To make a perfect single crystal, the first step is to melt the pure substance and then decrease the temperature very slowly until the desired crystal is produced. If the temperature is dropped too fast, the resulting crystal will have many defects. This procedure is known as **annealing**.

Simulated Annealing derived from Monte Carlo methods in statistical mechanics is a stochastic optimization algorithm that simulates the annealing process. Metropolis et al, first utilized the algorithm as a simulation method to examine the properties of substances consisting of interacting individual molecules [Ref. 1]. The purpose of the simulation is to find the ground states of a system which corresponds to the configurations of low energy molecular structure.

Kirkpatrick et al, and Cerny noticed that the search for low energy configurations in the annealing process could also

be linked to the search for low cost solutions in a **combinatorial optimization** problem [Ref. 3]. The problems related to the configurations of elements with a finite or countably infinite set are called **combinatorial problems**. Simulated Annealing shows a strong connection between statistical mechanics and combinatorial optimization.

According to the report, "Edge Detection by Cost Minimization" by Hin Leong Tan and Edward J. Delp [Ref. 4], a correspondence between statistical mechanics and combinatorial optimization can be drawn as in the following table.

Statistical Mechanics	Combinatorial Optimization
States(of system)	Solutions(to problem)
Energy(of state)	Cost(of solution)
Ground State	Optimal Solution

Sonar track data in image form is called a **Lofargram**; it shows the frequency spectrum (x-axis) of the received signal vs. time (y-axis). The problem of sonar track detection can be treated as an optimization problem, where the optimum among all possible curve positions is tried out on a Lofargram. Therefore the Simulated Annealing algorithm can be applied to detect the acoustic track among all possible state alternations.

A **state configuration** S is a two-dimensional array of pixels on an N × N square lattice:

$$S = \{ s(i,j) ; 1 \leq i,j \leq N \}$$

where each pixel can be a binary value 0 or 1. If s(i,j) = 1, then the pixel s(i,j) is an edge pixel [Ref. 4]. The state space of the annealing process is the set of all possible state configurations on an N x N square lattice. In this case, N is equal to 128. The state configuration in terms of a track line is a detection of the sonar track on the Lofargram.

B. Procedures of Simulated Annealing

The SA algorithm consists of the following steps:

(1) Select a random initial state $S_p = S_0$ and an initial temperature T_k, let k = 0.

(2) Randomly generate a next state S_n.

(3) Compare costs of two states S_n and S_p.

If $C(S_n) < C(S_p)$ then

transit to state S_n.

else

transit to state S_n with probability $e^{\frac{-[C(S_n)-C(S_p)]}{T_k}}$

(4) Increment k (decrement temperature) and then go to step (2).

(5) If the temperature drops approximately to zero, terminate the procedure.

C. Discussion

A random initial state and temperature schedule for the Simulated Annealing must be selected. At the start, the initial temperature must be high enough so that the progression of the new states is less dependent on their actual cost.

Theoretically, a potential next state should be generated according to a transition matrix of a Markov chain. However, there is no need to explicitly specify a transition matrix. Due to the tedious procedure, Hin Leong Tan concluded that "All that is needed is a method of generating next states such that certain conditions on irreducibility and reversibility are satisfied." [Ref. 4]

Given the present state S_p, a Markov chain is a random process where the next state S_{p+1} is dependent only on the immediately past states S_p. A Markov chain is irreducible if and only if all states can be reached from one another. The Simulated annealing process also has the property of weak reversibility; i.e., if state S_a can be transited from state S_b, then S_b can also be transited from S_a.

An appropriate cost function is used as a decision criterion for state transition. The cost function indicates how good the trial state is for the true position of the track which is to be detected. Selection of the cost function depends on the application problem. If only the next state which yields the lower cost solution is accepted as the next

current solution, the accepted current solution will be a local optimum. The aim however is to find the global optimum. This can be achieved by allowing an escape decision from the local optima randomly. Sometimes, it is necessary to climb a smaller hill to reach a lower valley. It is suggested to accept a higher cost of the cost function so that the cost function ridges can to be climbed over. Conceptually, the probability of being trapped in a local minima can be reduced. This decision process is not irregular either. The higher the hill, the less likely it is for the process to climb over the cost function hill. The acceptance probability is specified in step (3) of the previous section.

At each increment of k and decrement of temperature , a new next state is generated. Whether or not to transit to the next state depends on the rule in step (3). The above process is repeated until the cost function remains stable for a long time. This could be one kind of termination criteria. At the beginning of the process, the temperature should be decreased slowly since the state of the system is in an extremely random status. It is necessary to use small temperature stepsize to achieve the equilibrium as close as possible. When the annealing process approaches a steady state, the decrement of the temperature might then be increased to speed convergence.

Now that a general idea of Simulated Annealing has been introduced, three main issues needed to be considered for implementation. They are:

(1) How to generate the next state.

(2) How to define the cost function.

(3) How to control the cooling temperature.

The details of these implementation issues are be discussed in the next chapter.

III SINGLE TRACK DETECTION

This chapter describes the implementation of the SA algorithm discussed in Chapter II for the sonar track detection problem. For simplicity, a single track images is considered first. The experience gained from detecting a single sonar track can then be the basis of the multi-track detection problem described in the next chapter.

The problem of track detection involves locating the approximate position of the track and fine tuning to find the exact position. Initially, this thesis concentrates on the fine tuning process using SA. The track is confined to small region around the possible track. However, it is necessary to improve the technique beyond local optimization. To do this solution of the single track detection problem not only restricted to a small region but also allowed to a wider region. Three cases of this experiment are discussed according to their complexities.

(1) The starting point of the track is known.

(2) The starting point of the known track is within a small region of the true position.

(3) The starting point of the track is within a large region of the true position.

Case (1) is the simplest to solve. Even though case (1) is a simplistic case, the experience gained is quite useful to solve more complicated cases.

A. Case (1) : The starting point of the track is known

1. Algorithm of Simulated Annealing for detection of track.

a. State configuration

Based on the transition rule mentioned in chapter II about the procedures, the state configuration will remain unchanged or transit to the next state configuration. As the Simulated Annealing process proceeds, the state configuration will move in the state space. At the end of the process when a stable state is reached, the state configuration should approach the location corresponding to the true track in Lofargram.

b. Initial state generation

Although the track position j of the first line is known in a single track image, the next track position of the second line could be different. Each scan line of the lofargram contains one track pixel. The track pixels of the lofargram are connected to form a state configuration.

For the known track position in the first line, the track pixel of the second line can only be one of three positions. Suppose the track pixel is at column j in the first line. The track pixel in the second line can be only in position j-1, position j, or position j+1. To select a track position in

the second line of the initial configuration, the decision is based on a random number x,

$(0 \leq x \leq 1)$. The track pixel position on the second line vs. the probability distribution is shown in the following table.

random number x $0 \leq x \leq 1$	$0 < x \leq \frac{1}{3}$	$\frac{1}{3} < x \leq \frac{2}{3}$	$\frac{2}{3} < x \leq 1$
track pixel position of the 2nd line given the current track line is at j	column j-1	column j	column j+1

Different probabilities may be assigned to the table as desired. The positions of the track pixels for the remaining lines can be determined in a similar manner. This is how the initial state configuration is determined.

c. Next state generation

Theoretically, the next state should be chosen randomly and independently of the previous states. However, the choice for next state is not equal probable over the entire state space. States closer to the current one have higher probability. Therefore, it is easier to form a perturbation rule for generating the next state rather than a total random generation.

In this thesis, a strategy based on local perturbation for generation is used. The track pixel s_{ij} represents a track at scan line i, column j. The window $W_{i,j}(S)$ is the set of 9 pixels contained in a 3 × 3 region centered at pixel s(i,j):

$$W_{i,j}(S) = \{ s(m,n) : |m - i| \leq 1 \ \& \ |n - j| \leq 1 \}$$

s_{ij} is the center of the window $W(S_p)$. The neighborhood of a pixel $s(i,j) \in S$ is the set of 8 pixels specified by:

$$N_{i,j}(S) = \{ s(m,n) : |m - i| \leq 1 , |n - j| \leq 1 \ \& \ (m,n) \notin (i,j) \}$$

[Ref. 4]. Fig. 3.1 shows seven edge structures in $W(S_p)$ of the present state and their corresponding edge structures in $W(S_n)$ of the next state. If the edge structure in $W(S_p)$ belongs to one of the seven edge structures shown, the next edge structure in $W(S_n)$ is the structure shown on the right of the arrow. If none of the seven edge structures can be matched with $W(S_p)$, the state configuration at this step remains.

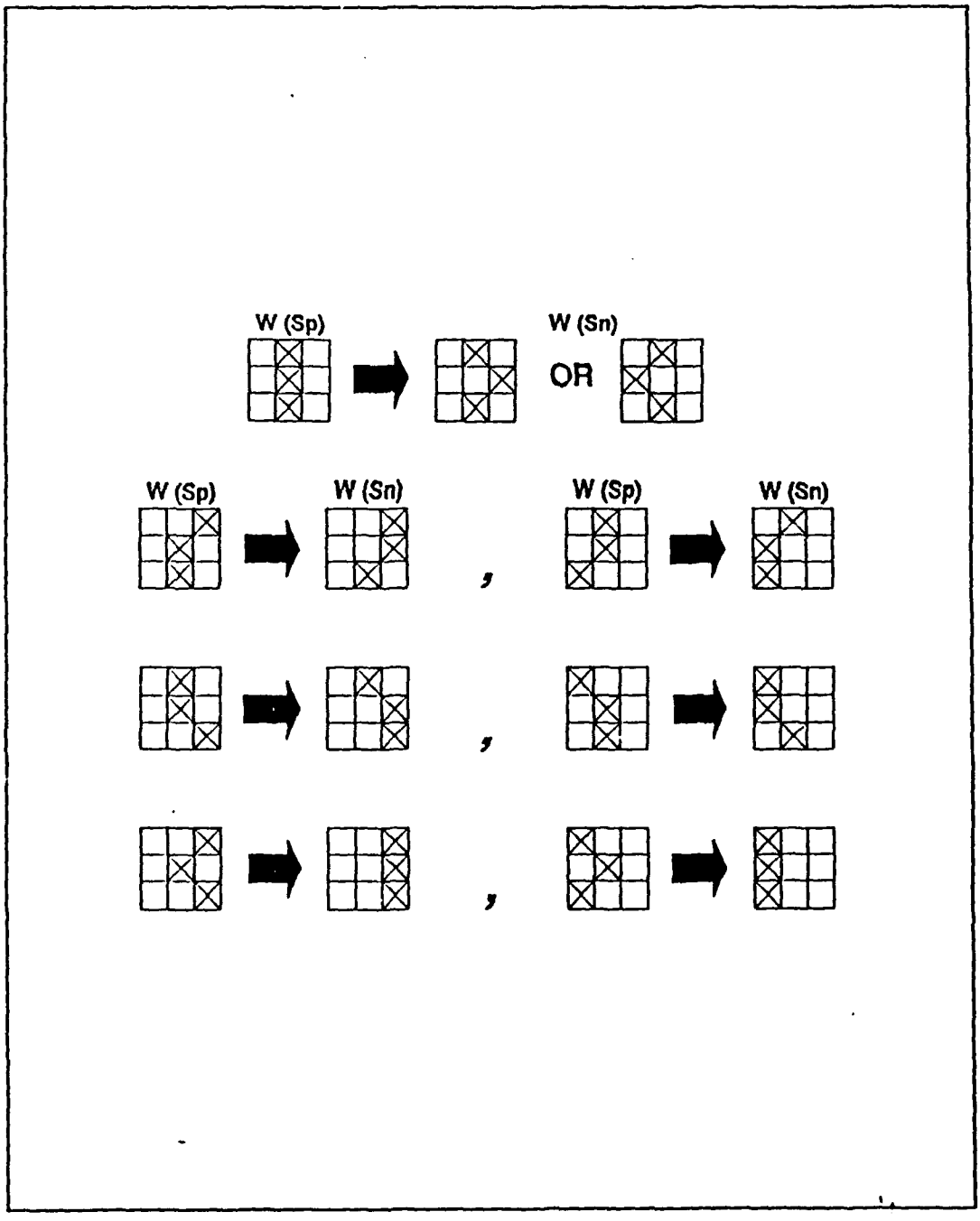


Figure 3.1 The seven edge structures in $W(S_p)$ and their corresponding $W(S_n)$ using a strategy of local perturbation.

d. Cost function and transition rule

Cost evaluation is necessary to decide whether the state S_p should transit to the state S_n or remain unchanged. If the edge structure of $W(S_p)$ matches with one of the seven edge structures shown, both the cost of S_p and the cost of the corresponding next state S_n are calculated. Then, a comparison of the costs $C(S_n)$ and $C(S_p)$, and a decision of whether S_p should change or not are done according to the rule given in Chapter II.

The cost function used in this thesis is a heuristic signal detection method involving an implicit threshold. The cost function involves both the track edge pixel information from the state configuration and the pixel intensity information from the image data. Once the temporal window has matched with a state structure, the corresponding sonar image pixel values are used to compute the cost. For a single scan line i the line cost function is defined as

$$C_i = \alpha\mu_i - P_i$$

where μ_i, average noise, is the mean of the pixel intensity of the region from the first column to the break of the ith line. The weight factor α is significant in thresholding, and P_i is the pixel intensity of the chosen break point of the ith line. A block contains three lines from line i-1 to line i+1. Therefore the block cost is

$$C_b = \sum_{l=i-1}^{i+1} C_l$$

The choice of the threshold parameter α is important. If the value chosen is too high, some valid tracks may be missed. But if the value is too low, false tracks may be created. According to the transition rule described in Chapter II, whether the state remains unchanged or transits to the next state depends on the factors shown in Fig. 3.2.

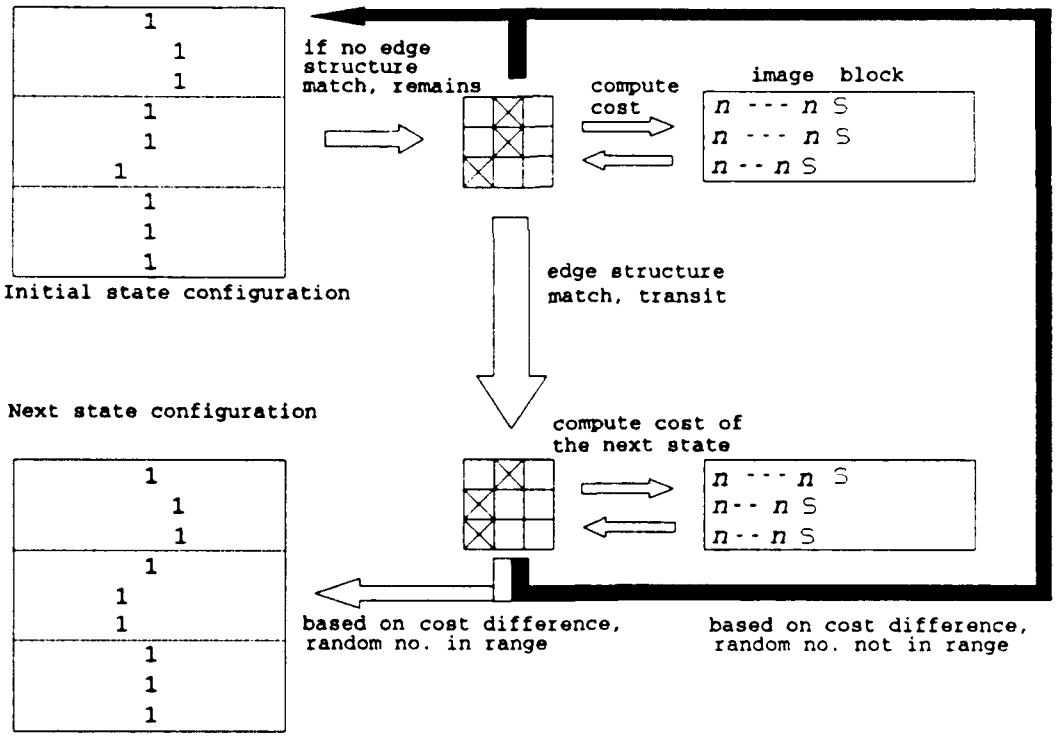


Figure 3.2 Illustration of the transition rule.

Notice that if a heuristic search algorithm is adopted for the next state transition, no random decision is necessary at the bottom of Fig. 3.2. In this case it usually does not lead to an optimum solution.

e. Temperature control

An appropriate and efficient cooling schedule (speed of temperature reduction) can cause the system to approach the equilibrium and reach the stable state (global optima) quickly. A common form used for the cooling schedule is

$$T(t) = \frac{T_H}{\log(t)}$$

However, if this schedule is used, run time can become very long. A more efficient (in the sense of run time) alternative form adopted for this thesis is

$$T(n) = \gamma^n T_H$$

where typical value for γ is 0.95. T_H must be high enough as explained in chapter II. The cooling schedule T_k plays an important role in the transition rule as the Simulated Annealing process goes on. Recall the transition rule:

If $C(S_n) < C(S_p)$ then

transit to state S_n.

else

transit to state S_n with probability $e^{\frac{-[C(S_n)-C(S_p)]}{T_k}}$

As the temperature is reduced, the occurrence of the lower cost state becomes more and more likely. If the rate of temperature reduction is too fast, it is unlikely that the system ends up in the global minimum cost state because of the lack of equilibrium. Only if the temperature is reduced slowly enough down to zero, is it possible for the system to reach the lowest cost state.

2. Experimental results

For case (1), eight sets of experimental data were used. They are artificially generated. The lofargram SNR values of these images are 3dB, 0dB, -3dB, -6dB, -9dB, -12dB, -15dB, and -18dB. The image sizes are 128 x 128 for all of the data. The original test images are shown in Fig. 3.3 and Fig. 3.4. Notice that for the data with -18dB SNR, the track is barely visible.

Given that the starting position of the track is located at 64 Hz, the initial state generated is shown in Fig. 3.5. This initial state is applied to image data sets of SNR = 3 dB and SNR = 0 dB, with $\alpha = 1.6$ and the number of iterations N = 200. The final results after Simulated Annealing process are the same in these two cases and are as shown in Fig. 3.5. The corresponding cost variation with the number of iterations for both sets of test data are shown in Fig. 3.6.

If a heuristic search algorithm is employed for the same test images, it is easy to be trapped to a local minimum. Use

the same test data with SNR = 3 dB, α = 1.6, and N = 200, the final result of the heuristic search shown in Fig. 3.7 is obviously unsatisfactory. Further, the cost reduction cannot reach a stable state.

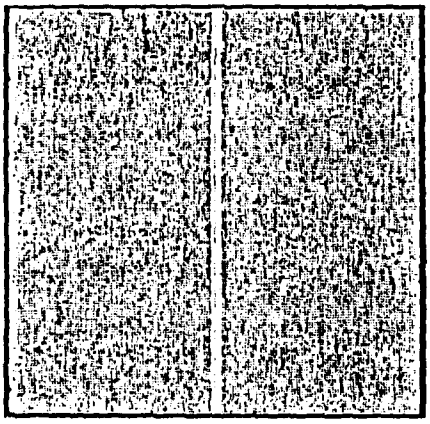
If the same value of α(= 1.6) is used for the test data with SNR = -3 dB, the result does not match the true track which is shown in Fig. 3.8. The cost cannot reach a stable state at the end of the simulated annealing process as shown in Fig. 3.9. By selecting a new value α = 1.8, the result becomes better as shown in Fig. 3.8. The cost can reach a stable state quickly in Fig. 3.9. Thus the selection of the value of α is somewhat data dependent.

The test data with SNR = -6 dB and -9 dB can reach the track position accurately in the same way as shown in Fig. 3.5. if α = 4.0 is used. Their costs can also reach stable states as shown in Fig. 3.10. This indicates that for lower SNR image data a larger α is desirable.

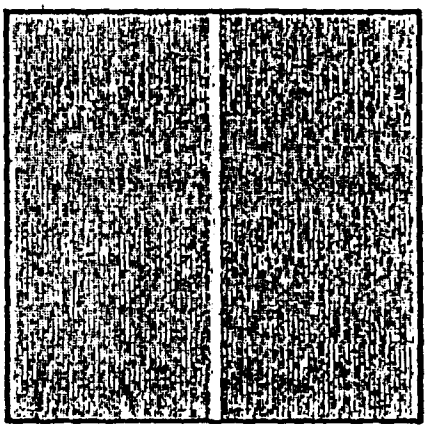
For the test data with SNR = -12 dB, an interesting result obtained. For α = 3.5 and N = 200, the final state is close to the true track position but not perfect as shown in Fig. 3.11. This suggests that the annealing process should be repeated several times(say 6 times) at each temperature so that equilibrium can be assured. In this case, the result is better slightly as shown in Fig. 3.11, but the cost still can not reach a stable state in Fig. 3.12. The cost cannot reach a stable state even if N is increased to 20,000 iterations as

shown in Fig. 3.12. There is so far no conclusive explanation of this result.

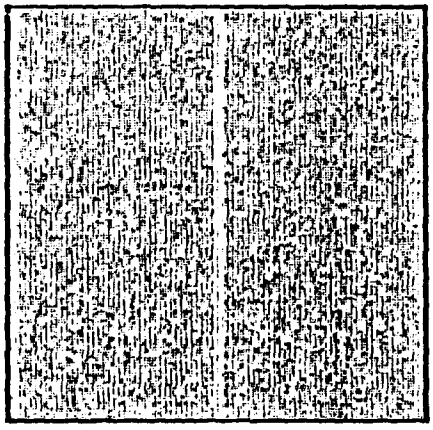
For test data sets with SNR = -15 dB and -18 dB, the necessity of repetitions at each temperature is even more obvious as shown in Fig. 3.13. Repetition at each temperature can more nearly assure the system equilibrium condition. These data sets are run with and without repetition at a given temperature, but the total number of iterations is the same. The corresponding cost function values for both approaches are shown in Fig. 3.14. Notice that for the test data with SNR = -18 dB, the number of repetitions has to be at least 9 to yield a better result.



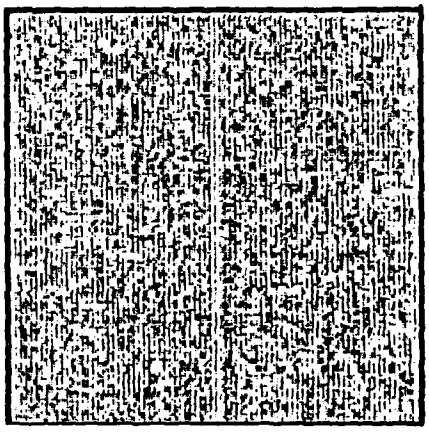
3 dB



0 dB

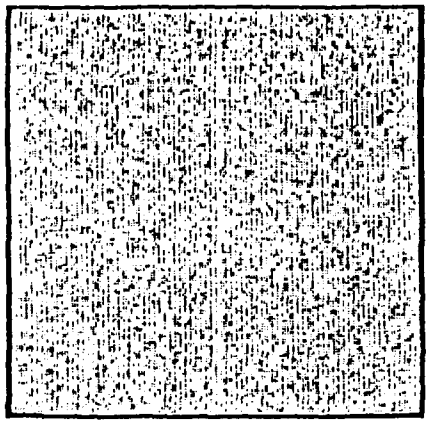


-3 dB

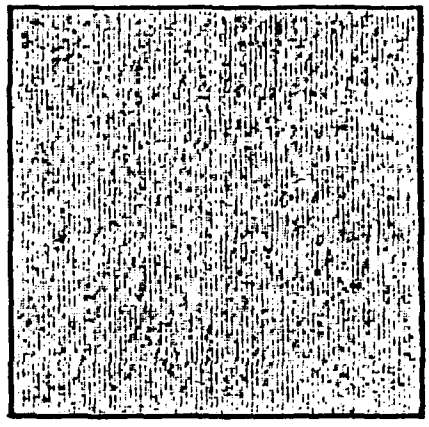


-6 dB

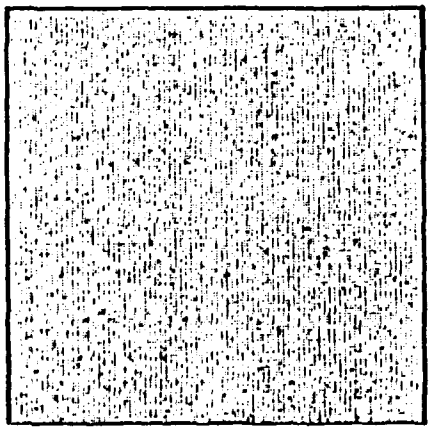
Figure 3.3 Original image with SNR from 3 dB to -6 dB.



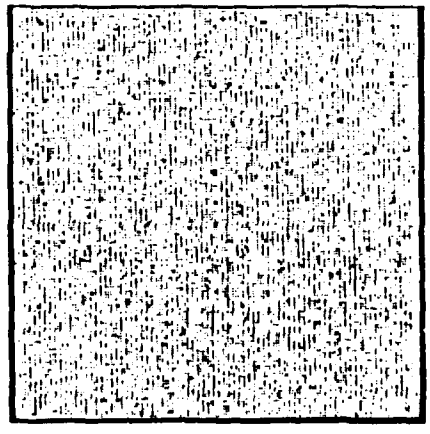
-9 dB



-12 dB

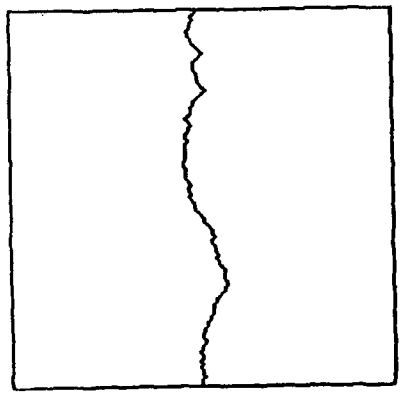


-15 dB

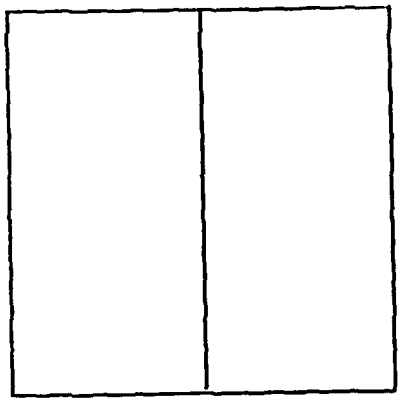


-18 dB

Figure 3.4 Original image with SNR from -9 dB to -18 dB.

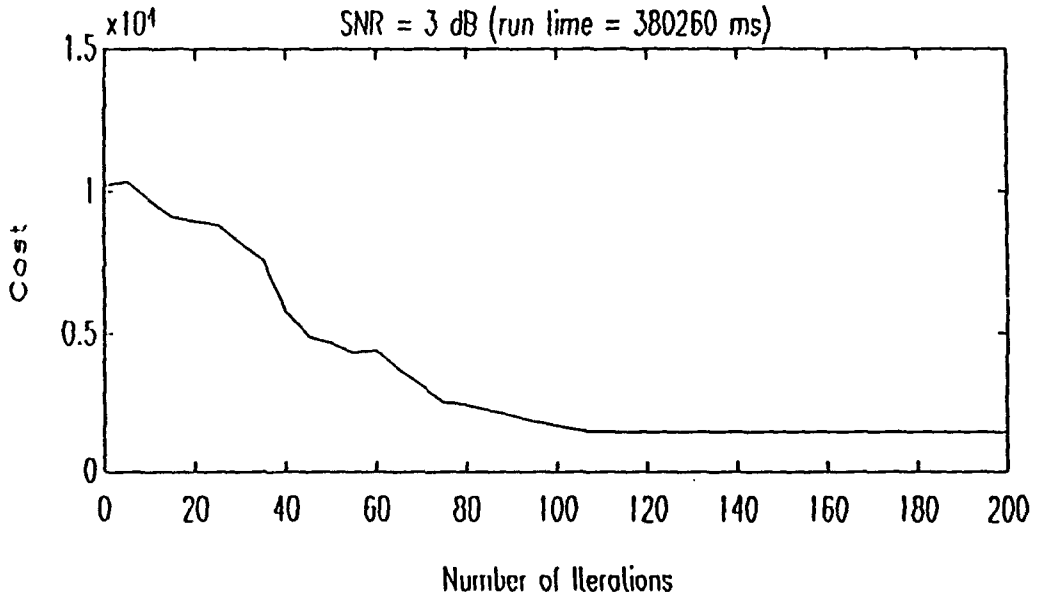


Initial state



Same final state for
SNR = 3 dB & 0 dB
(α = 1.6, N = 200)

Figure 3.5 Initial state and the result where the track starts at 64 Hz.



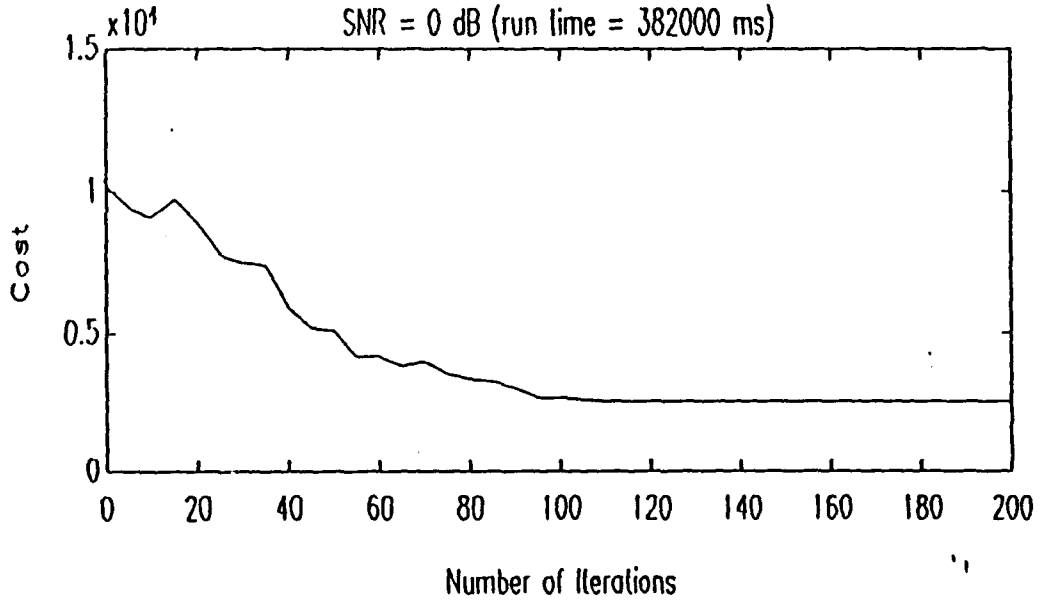
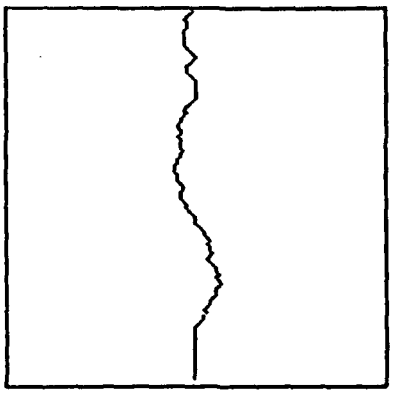


Figure 3.6 The corresponding cost variation for test data of SNR = 3 db and 0 dB.



Final state using heuristic search

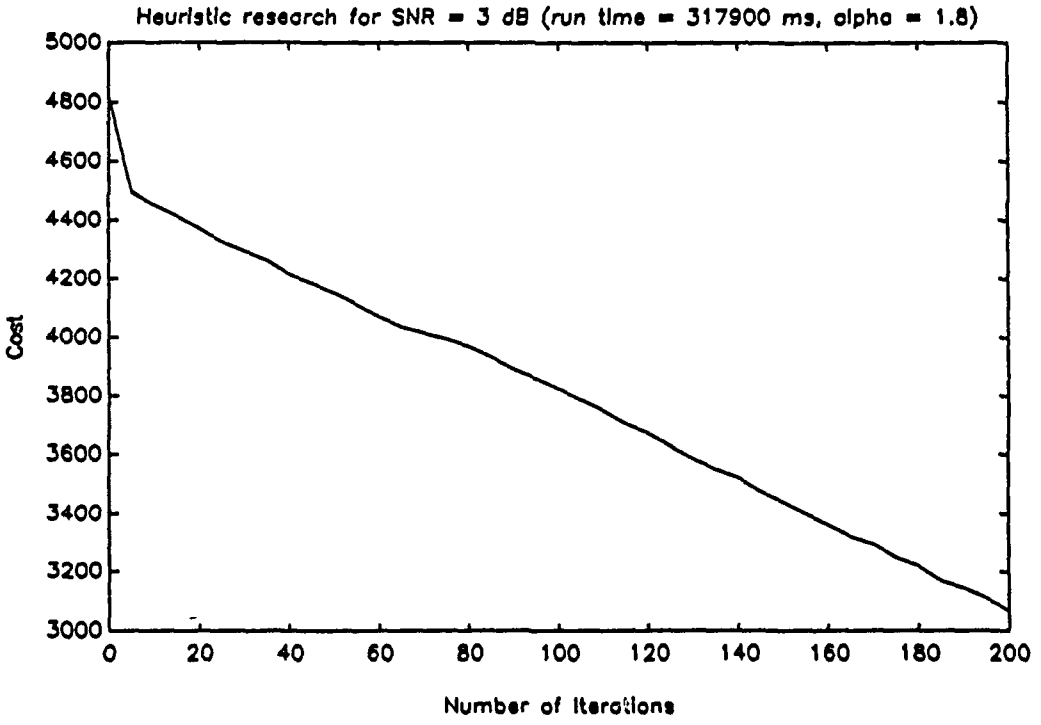
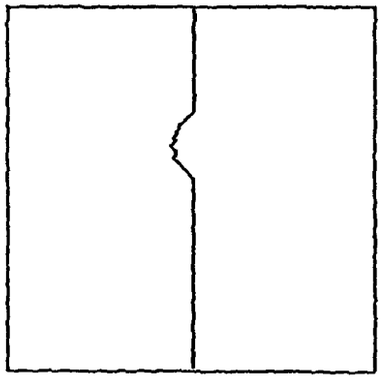
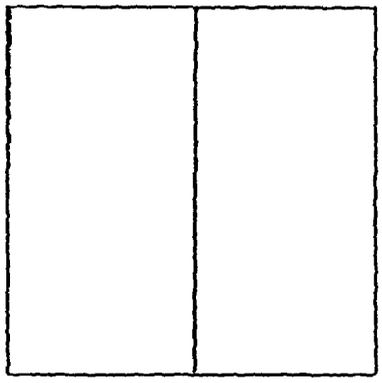


Figure 3.7 The result and the cost of the heuristic search algorithm for test data of SNR = 3 dB



Final state for
SNR = -3 dB
(α = 1.6, N = 200)



Final state for
SNR = -3 dB
(α = 1.8, N = 200)

Figure 3.8 Same image data with different α value.

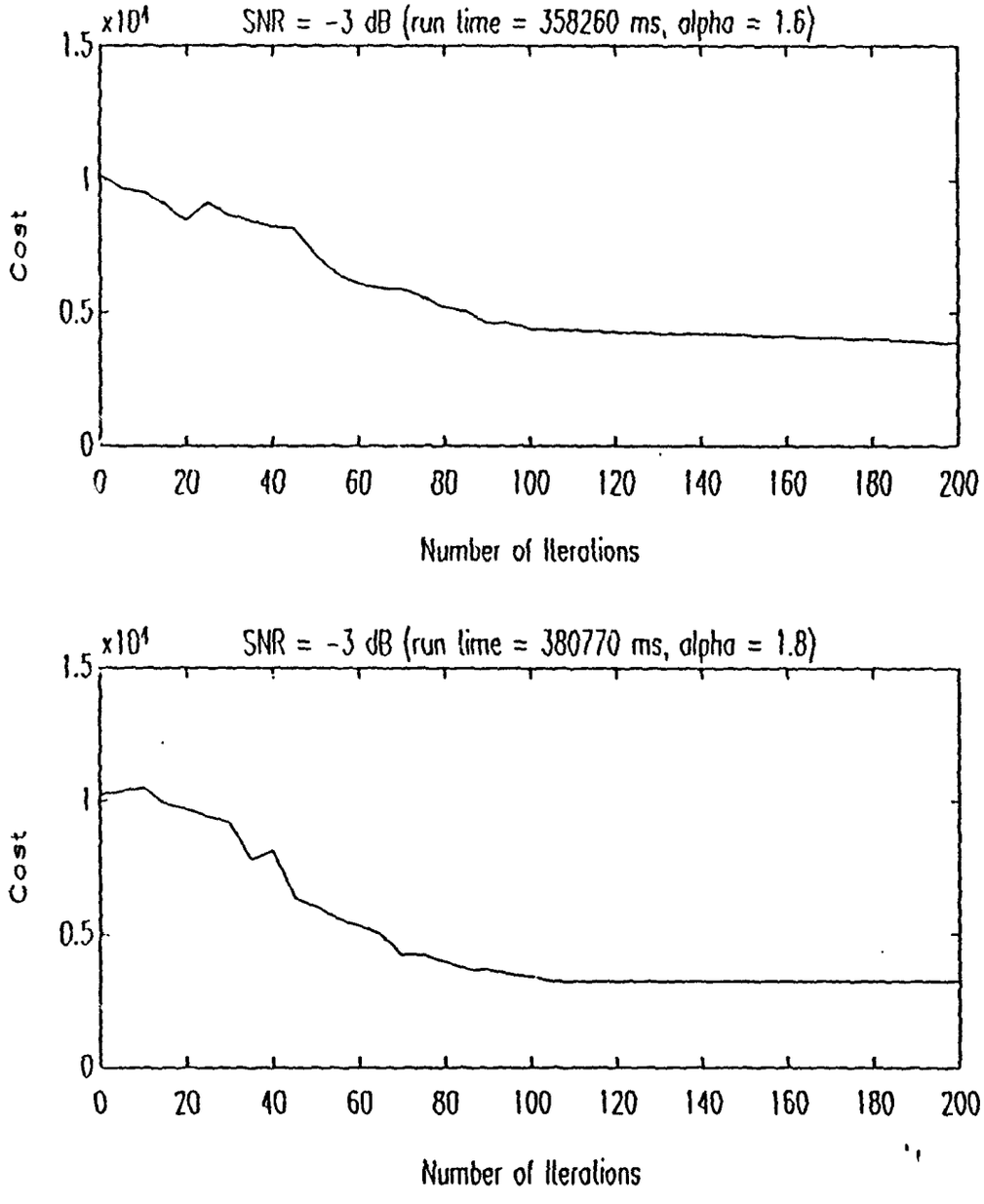
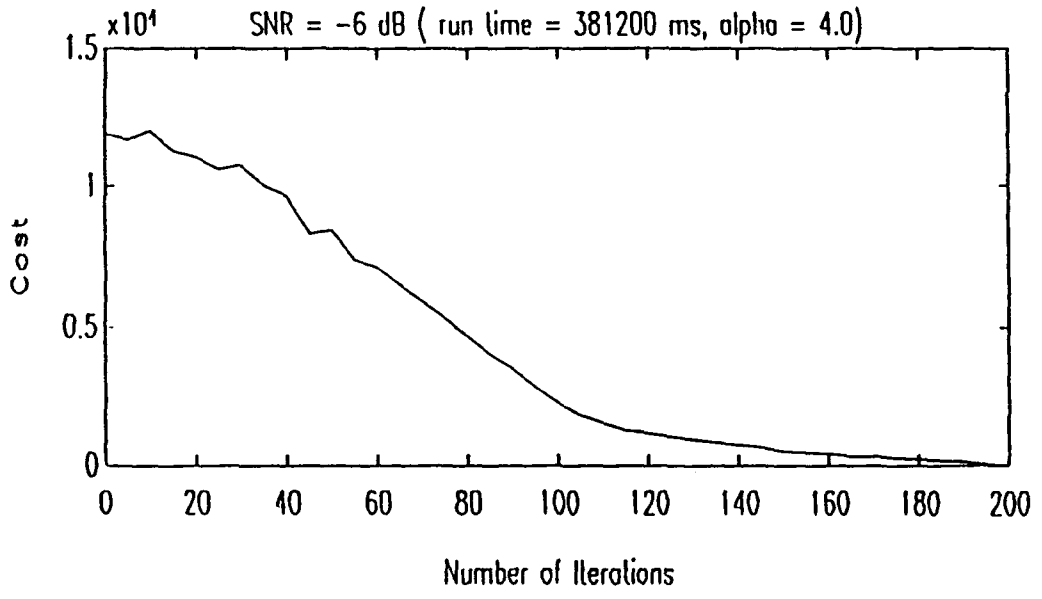


Figure 3.9 The cost variations for the same image data with different α value.



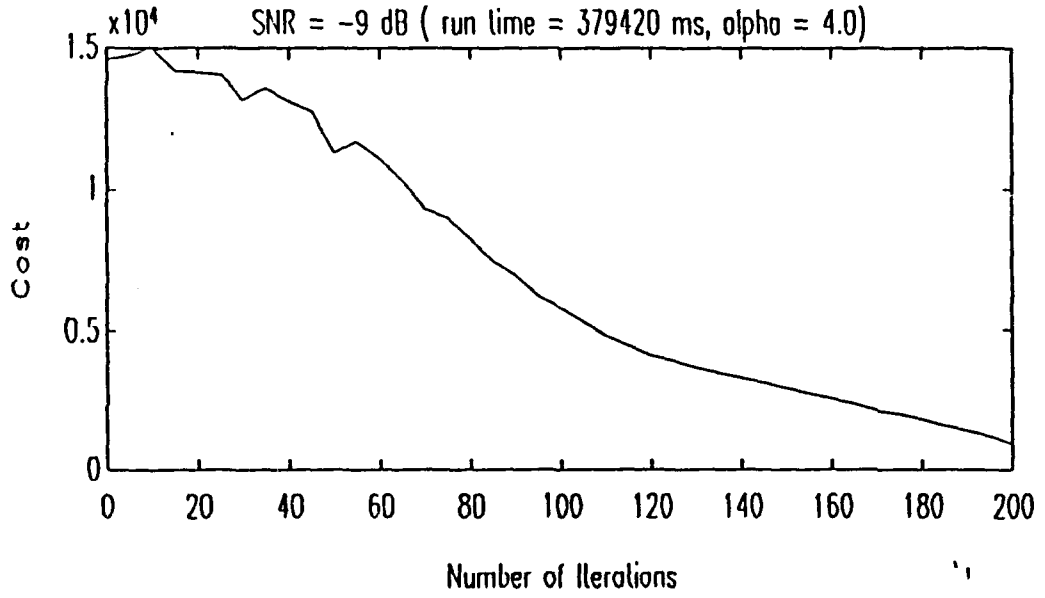
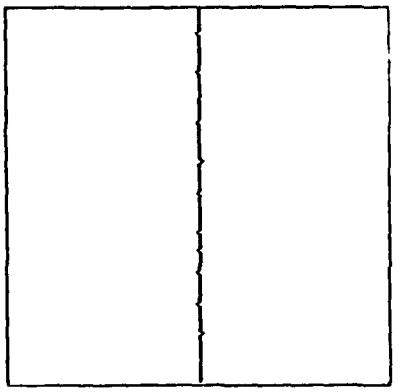
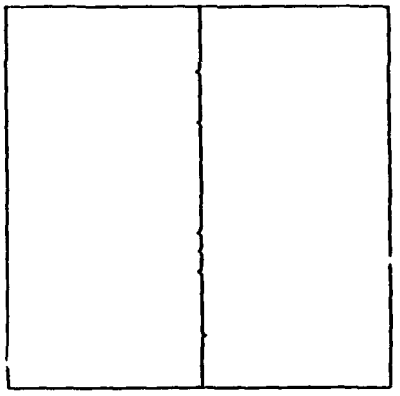


Figure 3.10 The cost variations for image data of SNR = -6 dB and SNR = -9 dB.



Final state for
SNR = -12 dB
(α = 3.5, N = 200)



Final state for
SNR = -12 dB
repeat 6 times
at each temperature
(α = 3.5, N = 200)

Figure 3.11 The results for image data of SNR = -12 dB without and with 6 repetitions per temperature.

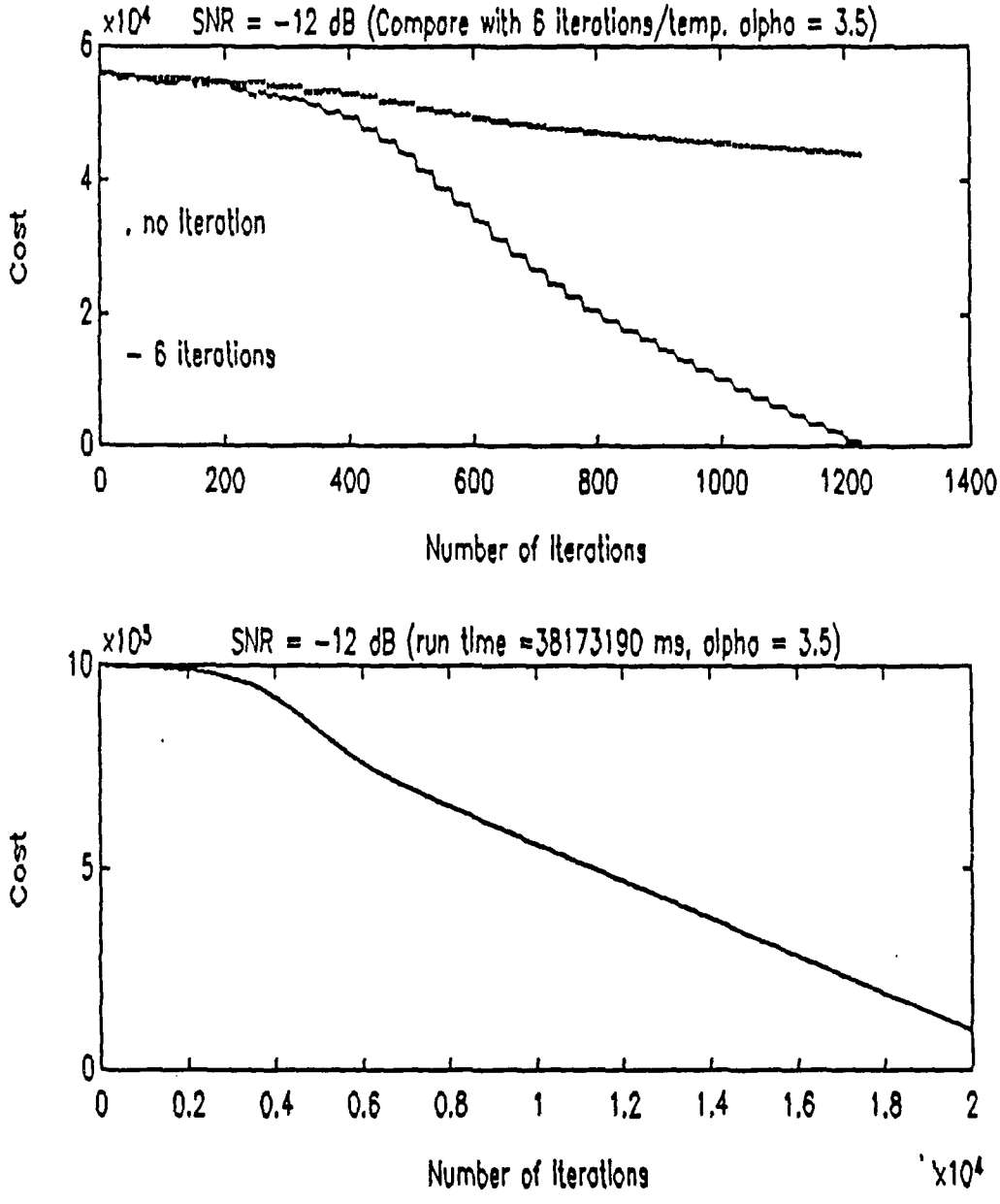
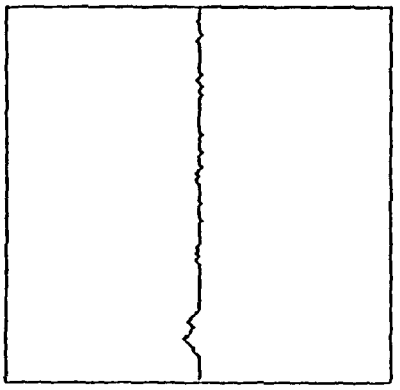
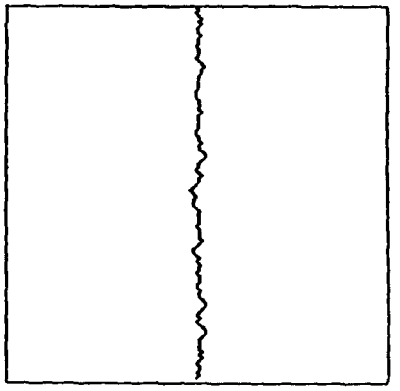


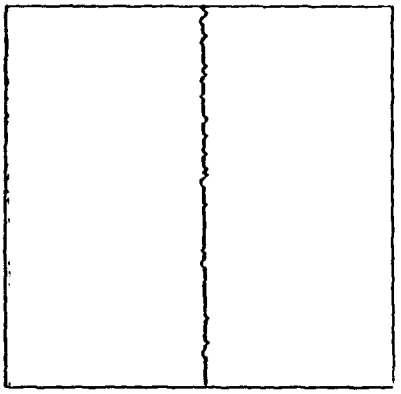
Figure 3.12 The cost variations for image data of SNR = -12 dB.



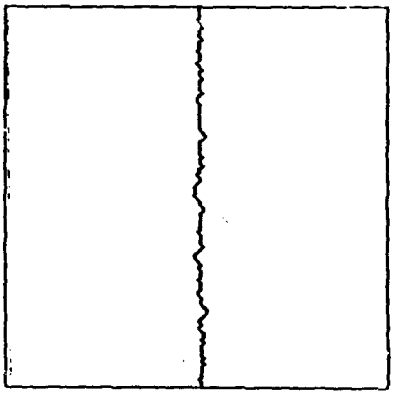
Final state for
SNR = -15 dB
(α = 3.5, N = 1200)



Final state for
SNR = -18 dB
(α = 3.5, N = 1800)

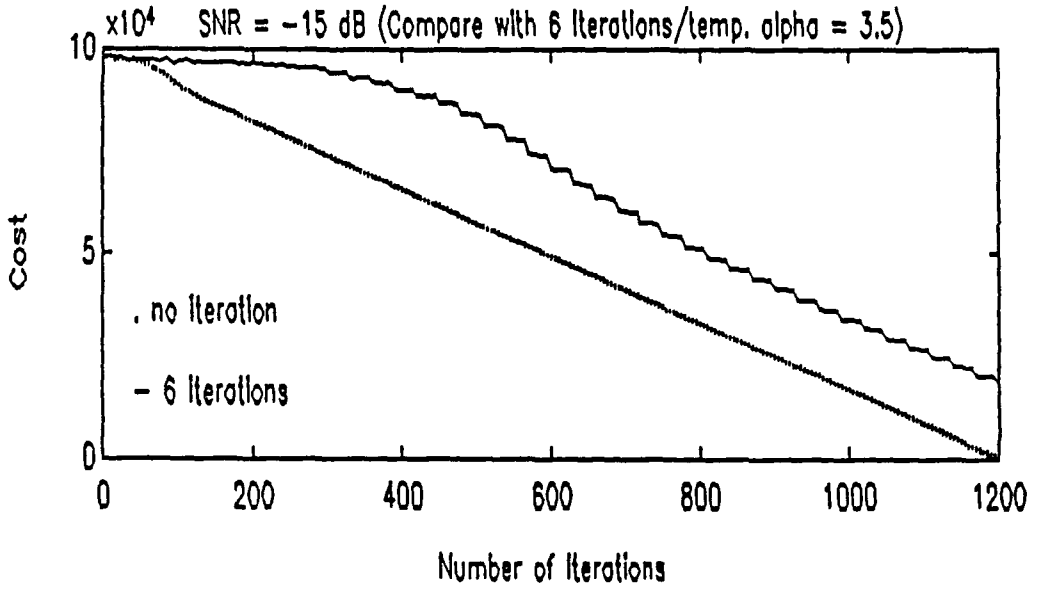


Final state for
SNR = -15 dB
6 repetitions/temp.
(α = 3.5, N = 200)



Final state for
SNR = -18 dB
9 repetitions/temp.
(α = 3.5, N = 200)

Figure 3.13 The results for image data of
SNR = -15 dB and SNR = -18 dB.



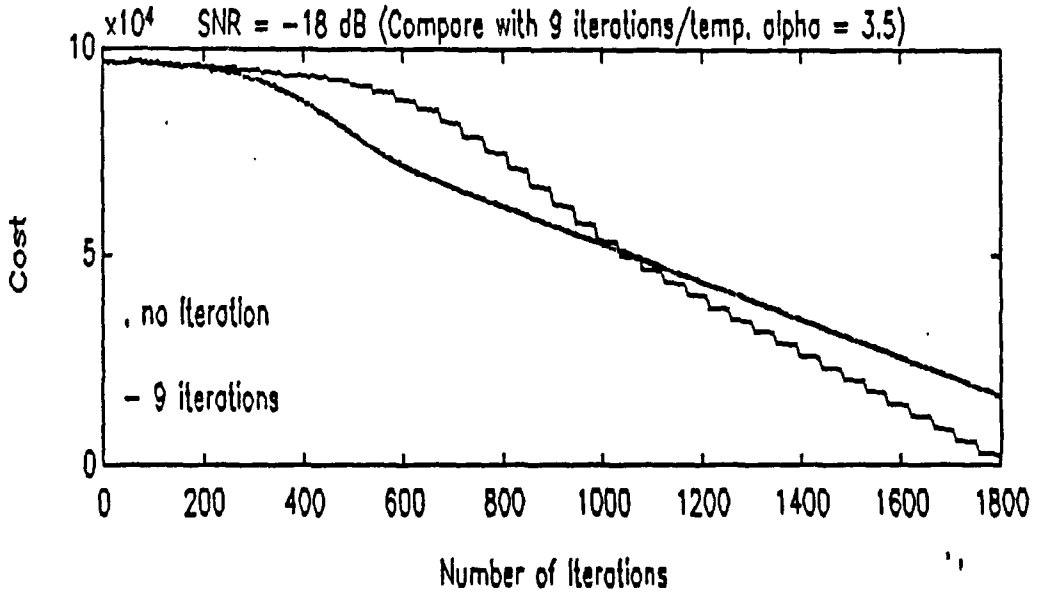


Figure 3.14 The cost variations for image data of SNR = -15 dB and SNR = -18 dB.

B. Case (2) : The starting position of the track is within a small region of the true track

1. Algorithm

a. Forward and reverse scanning

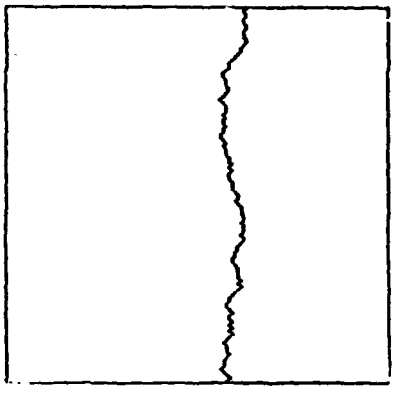
Suppose the exact starting position of the track is unknown, but the starting position is known to be within a small region around the true track position. The approach adopted here is to arbitrarily select a first track position and generate the initial state as before. Utilizing the same algorithm in case (1), the result has a fair chance to match the lower portion of the true track. Because the starting position is not allowed to move, the position of the result which is close to the starting position cannot reach the true track unless the chosen starting position of the initial state happens to be correct. Since the lower part of the final state is reliable, the end point of the result may be adopted as the starting position of a new "initial state". The next thing to do is to use the same algorithm in case(1), but change the scan and reverse the order from bottom to top.

b. Unconstrained starting position

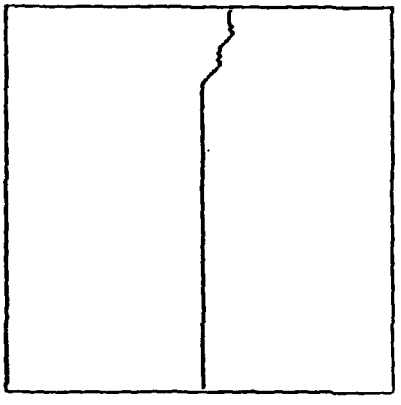
Another simpler idea is to liberate the starting point. Let the starting point move along the tendency of the track pixel in the second scan line. This procedure also has a good chance to produce the final state configuration close to the starting position of the track.

2. Experimental results

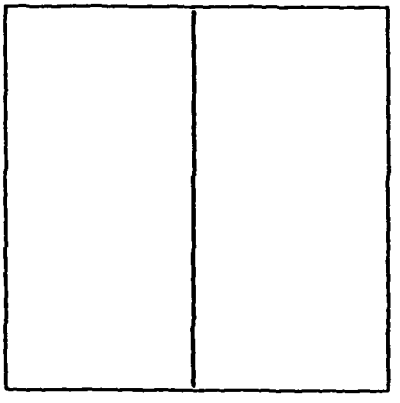
To illustrate the results of the above algorithms, a data set with SNR = 3 dB has been used. The initial state starting at 75 Hz is arbitrarily chosen in Fig. 3.15. The final result in Fig. 3.15 shows that a very large portion of the track matches the true track. This result is adopted as the new "initial state" and the same algorithm is used in the reverse scan order from bottom to top. The result in Fig. 3.15 shows a good match to the true track. The costs of both procedures, forward and reverse scanning, are shown in Fig. 3.16. The result of unconstrained first state position using the same data and initial state is also good. The result and its corresponding cost are shown in Fig. 3.17.



First state for
SNR = 3 dB
start from 75 Hz
(α = 1.6, N = 200)



Final state for
SNR = 3 dB
of front scanning
(α = 1.6, N = 200)



Final state for
SNR = 3 dB
of reverse scanning
(α = 1.6, N = 200)

Figure 3.15 The result using forward and reverse scanning.

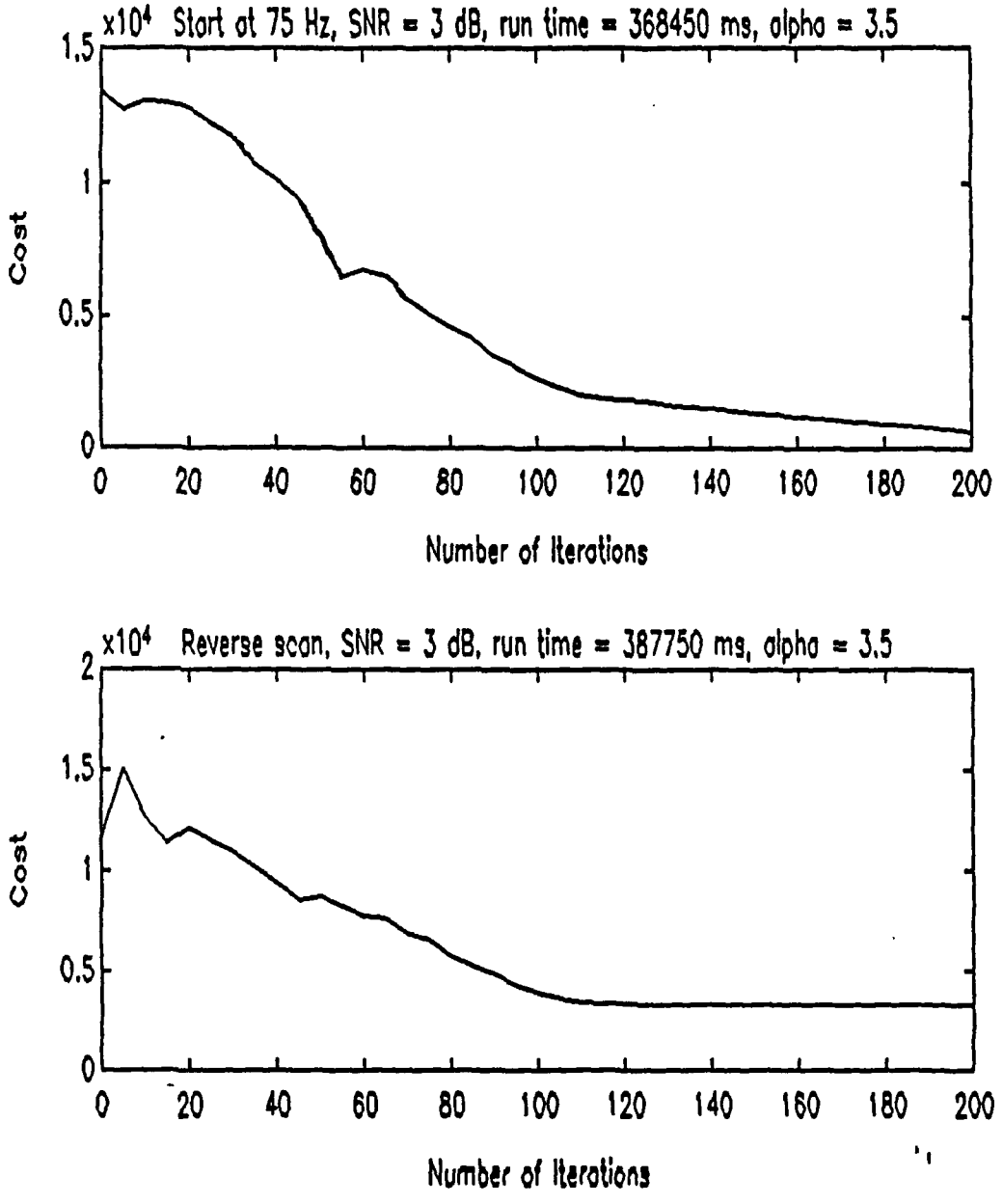
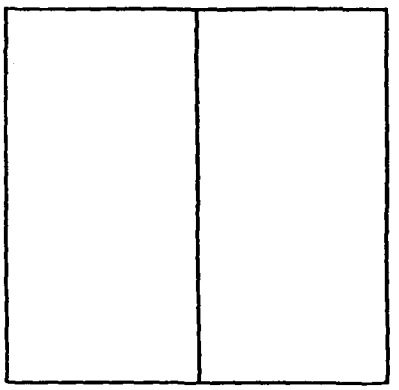


Figure 3.16 The cost variations of both the forward and the reverse scanning procedures.



Final state for
SNR = 3 dB of
liberating the
starting point

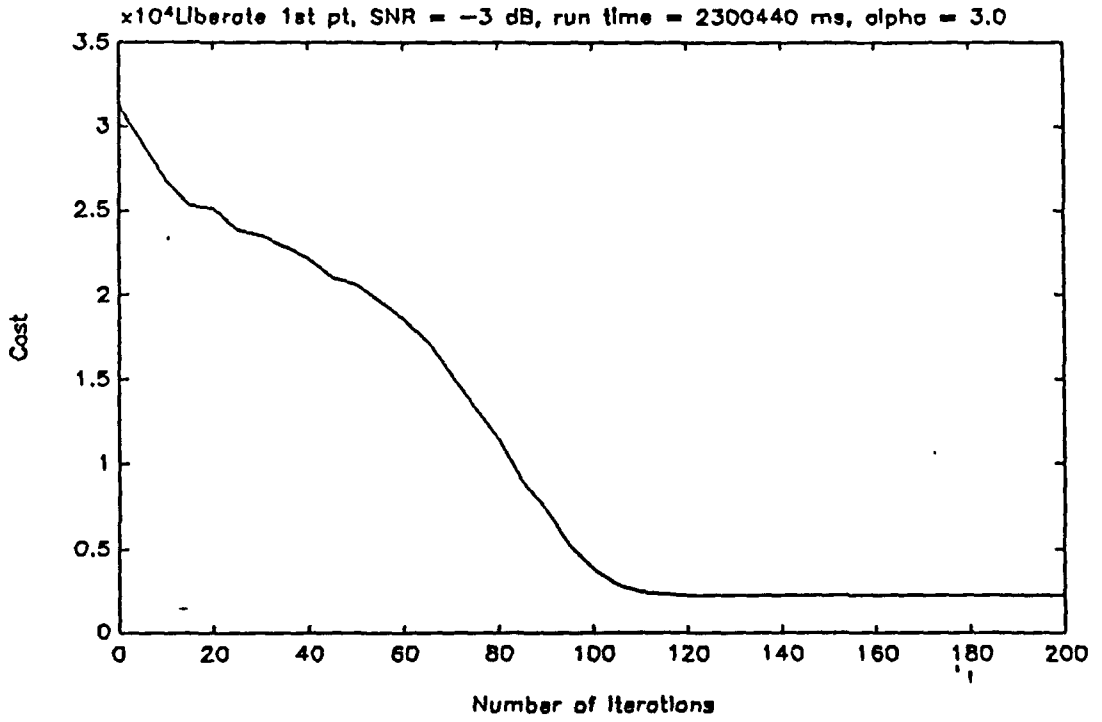


Figure 3.17 The result and its corresponding cost of unconstrained first state position.

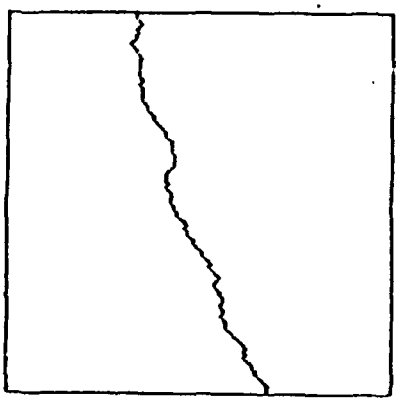
C. Case (3) : The starting position of the track is within a large region of the true track

1. Algorithm

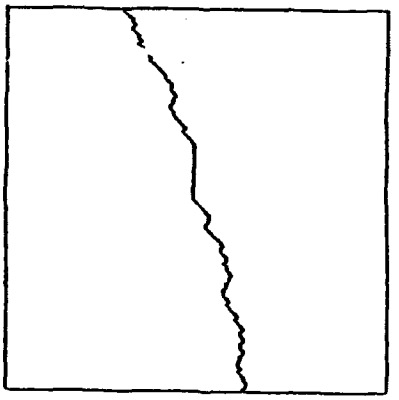
The initial state chosen in this case is intended to swing over a wider range of frequency spectrum. If the initial state intersects with some portion of the true track somewhere, there is a good chance that the result will follow the true track in the small overlapping region. The result covers a small part of the search region. Take the final state as a new "first state" and repeat the Simulated Annealing procedure. If the matched region is lengthened, it assures that the previous result is not a coincidence. The final state should shrink to a smaller strip of the lofargram. This procedure is repeated until the final state eventually reaches the track.

2. Experimental results

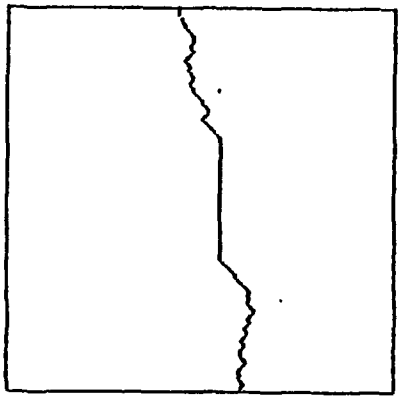
The data with SNR = -3 dB is used. The first state generated which starts from 40 Hz to 80 Hz is shown in Fig. 3.18. The final state shows a short match in Fig. 3.18. If the final state is taken as a new "first state" and the same procedure is repeated, the result in Fig. 3.18 shows that the matched segment extends. To reach the track faster, six iterations per temperature setting were used for the fourth repetition. The results shown in Fig. 3.19 is still good. The corresponding costs are shown in Figures 3.20 and 3.21.



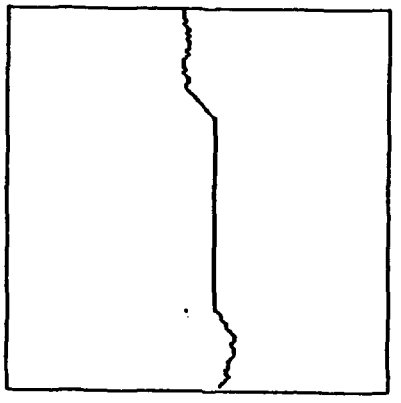
Initial state



Final state after
1st processing

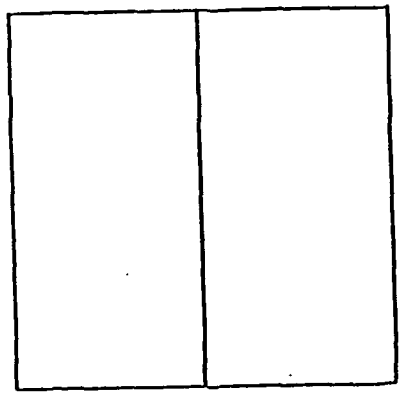


Final state after
2nd processing



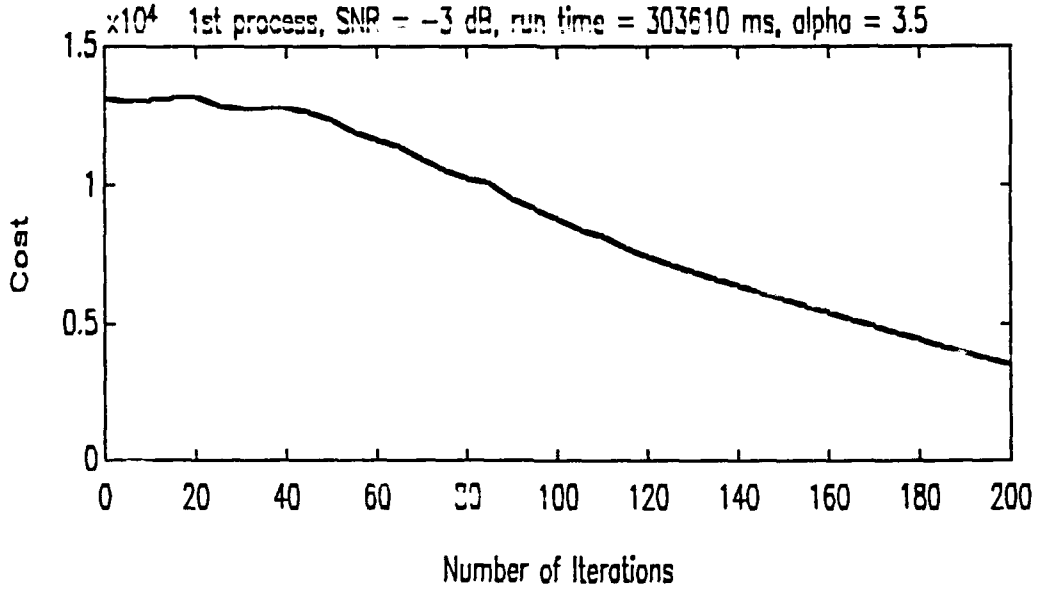
Final state after
3rd processing

Figure 3.18 The results of unknown tracking position after each process.



Final state after
4th processing
6 repetitions/temp.

Figure 3.19 The final result after 4 processings.



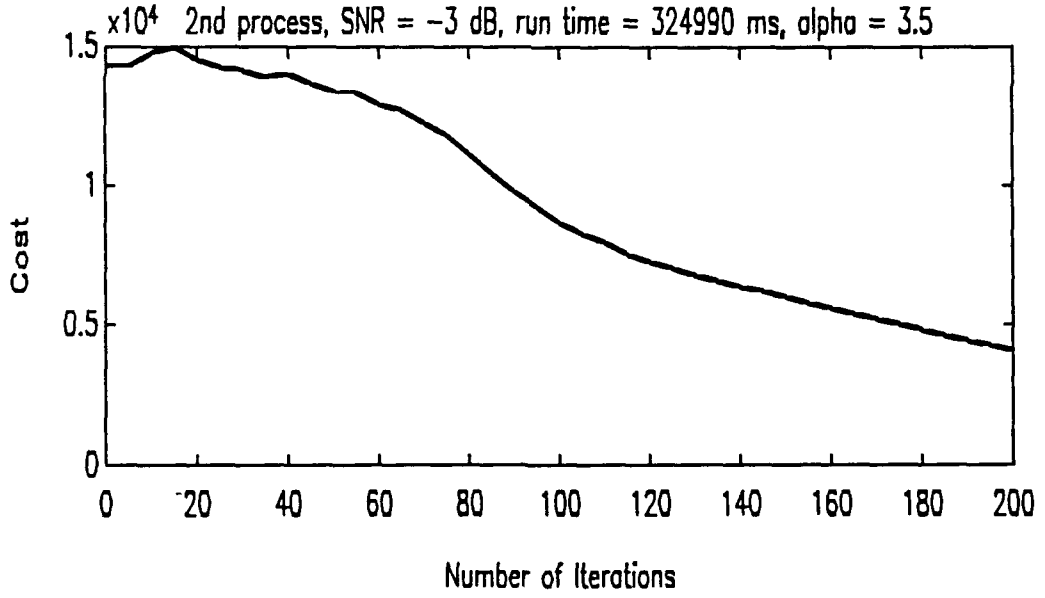
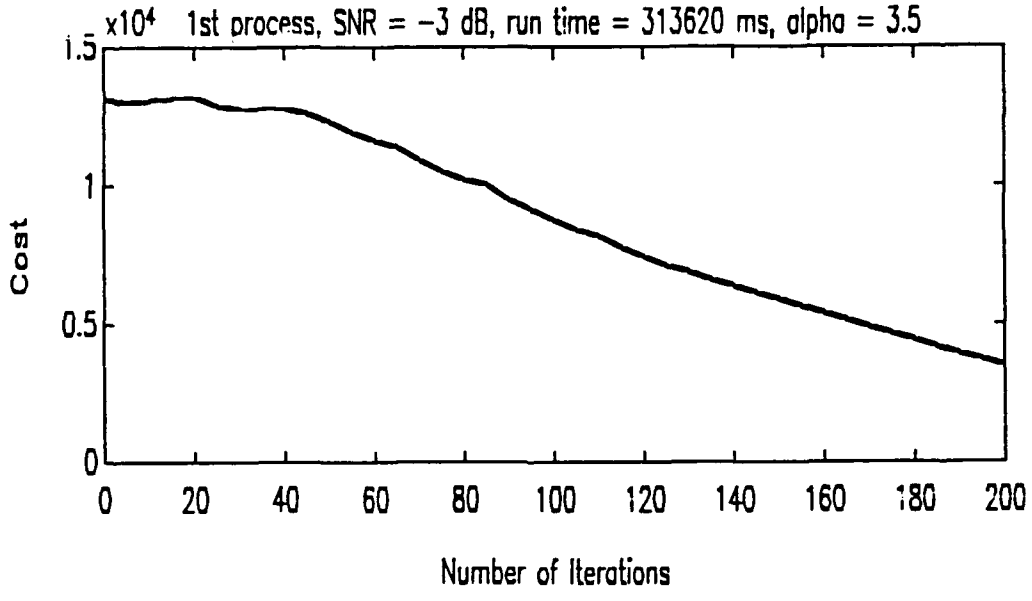


Figure 3.20 The cost variations of the first two processes.



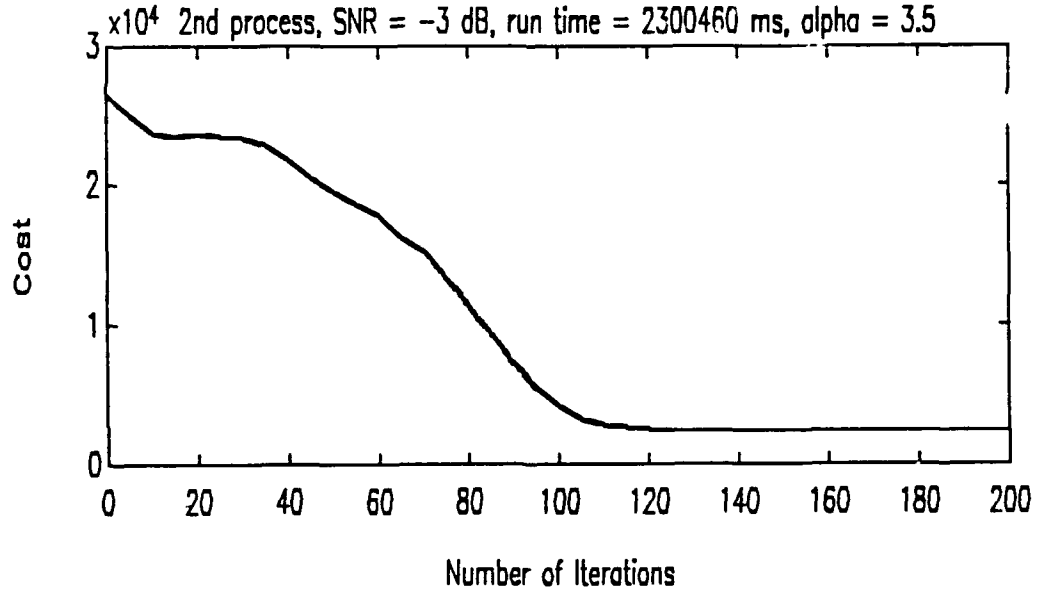


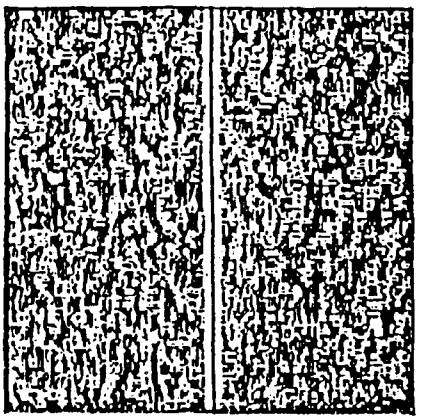
Figure 3.21 The cost variations of the third & fourth processes.

D. Comparison with classical edge detectors

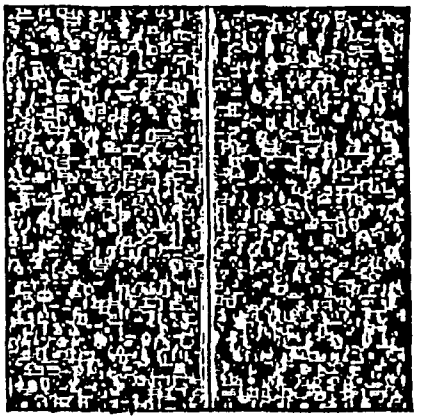
The proceeding two sets of experimental results show that the Simulated Annealing algorithm can detect a noisy sonar track with very low SNR value. For comparison, some results of classical edge detectors are shown to demonstrate the advantage of the SA algorithm. Fig. 3.22, 3.23, and 3.24 show the results of the same experimental data sets using Sobel, Roberts, and Laplacian edge detectors. None of them can detect the track with SNR below -15 dB. This is because they are all too sensitive to noise.

E. Concluding remarks

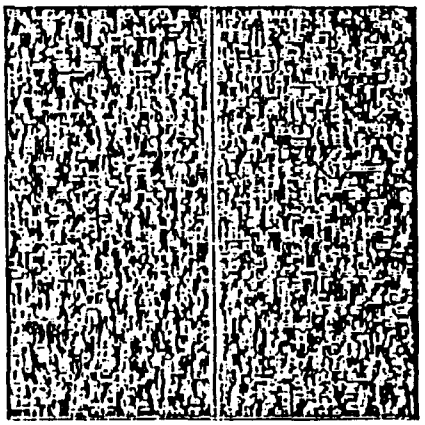
In this chapter the SA algorithm was implemented for the problem of single track detection. The experimental results show that the sonar track can be detected down to a very low SNR where most classical edge detectors do not work at all. The experience gained here is useful to help solving the problem of multi-track detection in the next chapter.



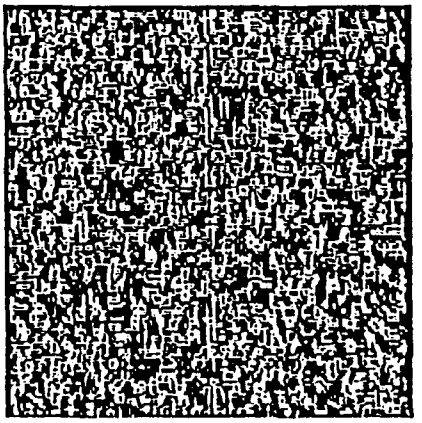
SNR = 3 dB



SNR = -3 dB

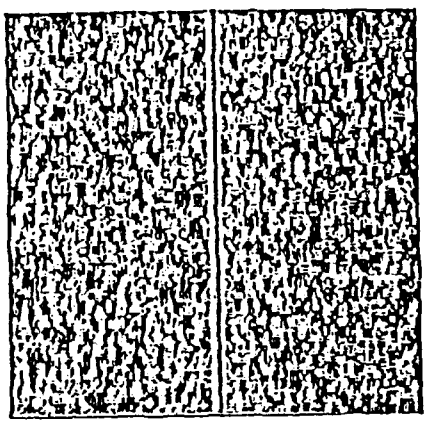


SNR = -9 dB

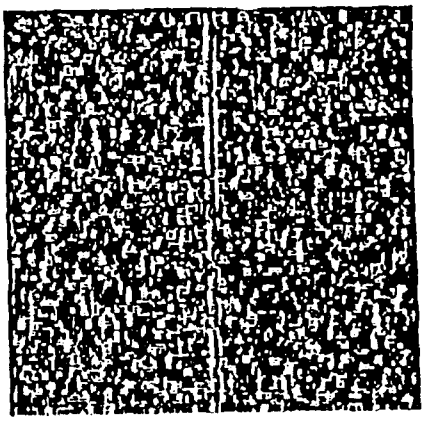


SNR = -15 dB

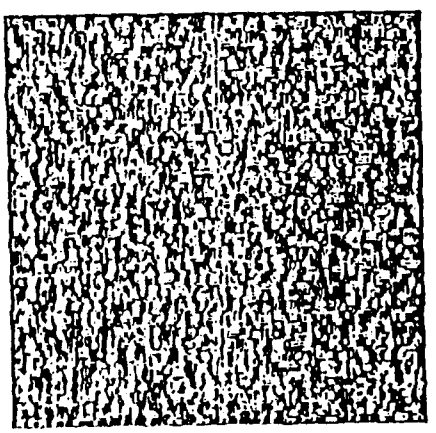
Figure 3.22 The results using the Sobel edge operator.



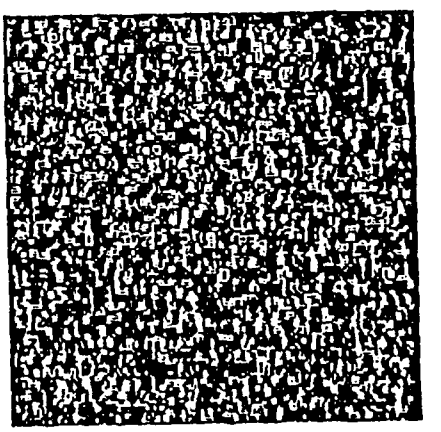
SNR = 3 dB



SNR = -3 dB

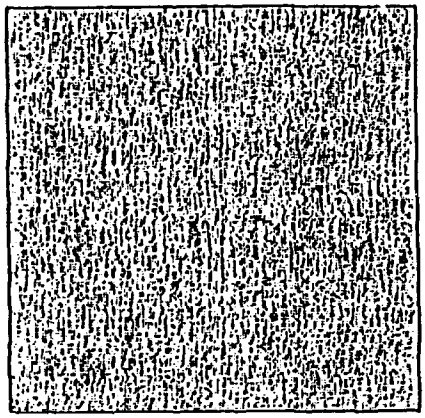


SNR = -9 dB

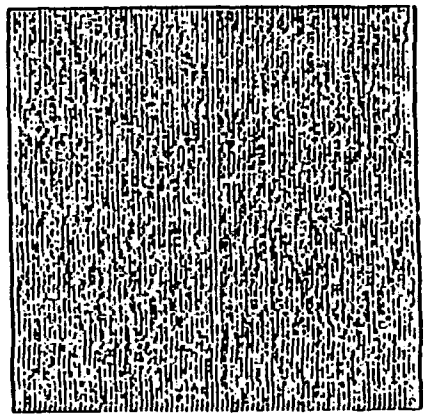


SNR = -15 dB

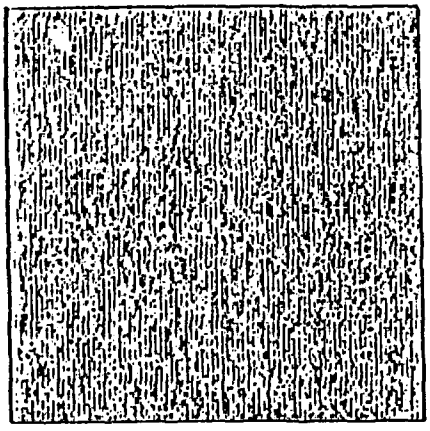
Figure 3.23 The results using the Roberts edge operator.



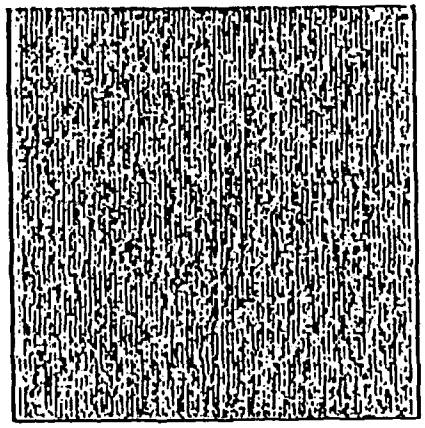
SNR = 3 dB



SNR = -3 dB



SNR = -9 dB



SNR = -15 dB

Figure 3.24 The results using the Laplacian edge operator.

IV MULTITRACK AND SWEEP TRACK DETECTION

This chapter uses the experience gained in the previous chapter as the basis of solving the multi-track detection problem. The experimental image data used here includes two types of multi-tracks images, images containing three-straight tracks and images containing double-sweep tracks.

A. Algorithm

The algorithm is basically the same as the one used in the single track detection problem, with some modifications. The initial state and the next state are generated in the same way as that of the single track detection, except that now three track lines need to be generated in the initial state. The main difference is in the method of transition from the initial state to the next state. For a single track detection problem, there is only one track pixel on each scan line. Therefore, only one decision has to be made for the position of the track pixel on each line. But for the multi-track detection problem, there is more than one track pixel on each scan line. Several decisions have to be made for the positions of all track pixels on each scan line.

For a multi-track detection problem, the state transitions of all track pixels on each scan line must be determined at the same time. At scan line i, a state block which includes

scan line i-1 to line i+1 has to be stored in a temporary memory to calculate the block cost of the initial state. Then the same strategy of local perturbation as in Chapter III is used to generate the next state. If none of the seven edge structures can be matched with $W(S_p)$ of any of track pixels and its neighbors, the state configuration for line i remains. Otherwise, a transition $W(S_p)$ to its corresponding $W(S_n)$ is made and the block cost of the next state is calculated. The track pixels on scan line i will either transit to the next state together at the same time or all remain at the same positions, according to the transition rule. The transitions of the track pixels on each scan line cannot be considered separately.

To make the track pixels transit altogether may cause some track pixels to move to higher cost positions. But, more track pixels can also transit to lower cost positions. The overall cost should eventually decrease to a steady state through the annealing process. The run time may be increased in this way, but the system is closer to equilibrium.

The transition rule for the multi-tracks is shown in Fig. 4.1.

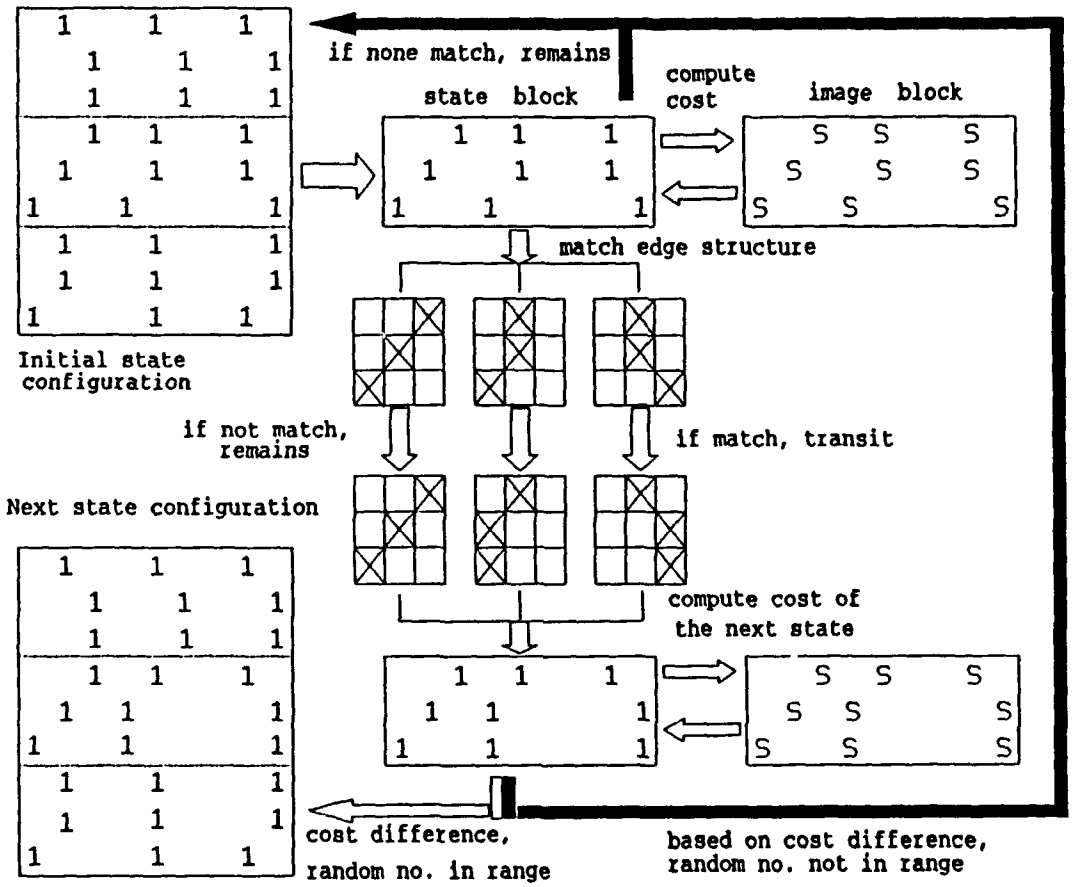


Figure 4.1 Illustration of the transition rule for multi-track detection.

B. Experimental results

Two sets of experimental data have been used. The first one is a three-track image with SNR = -12 dB. The three tracks are constant sinusoids of frequencies 30 Hz, 60 Hz, and 90 Hz. The original image and the initial state are shown in Fig. 4.2. A straight Simulated Annealing process for 1800 iterations produces a deviated result as shown in Fig. 4.3. Using nine repetitions per temperature in SA process for 200 iterations, result in the improved result of Fig. 4.3.

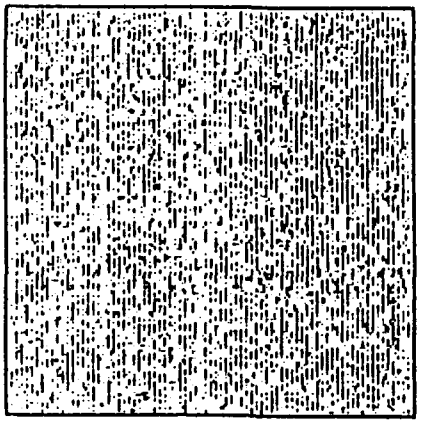
The second set of image data consists of the sweep double-tracks in Fig. 4.4. The sweep tracks which result from the Doppler effect represent that the target is approaching or leaving from the sensors. The two tracks start separately at 8 Hz and 74 Hz. The first track at 8 Hz has very weak intensity. The initial state and the result after the annealing process are also shown in Fig. 4.4. Notice that the tracks do not swing over a wide range of frequency due to the speed limit of ships . The doppler effect is usually not strong. Thus the sweep tracks data very similar to the three-tracks data set.

The cost variations for both data sets are shown in Fig. 4.5.

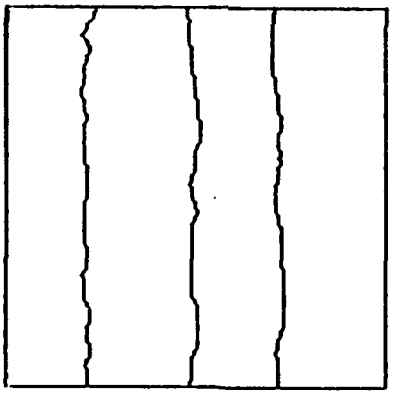
C. Comparison with classical edge detectors.

Again, three classical edge detectors were used to detect the three-tracks and the sweep-tracks. The results as shown in Figures 4.6 and 4.7 are very poor. Notice that the pixel

intensity of each track is weaker than the single track although the average SNR is the same as the SNR of the single track.

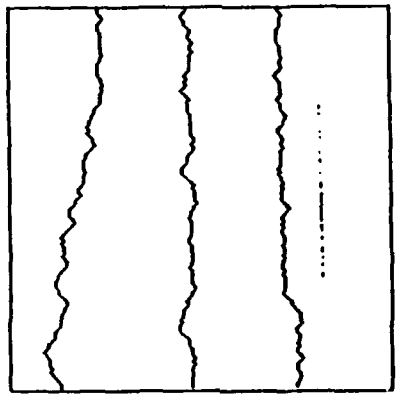


Three-track image
SNR = -12 dB

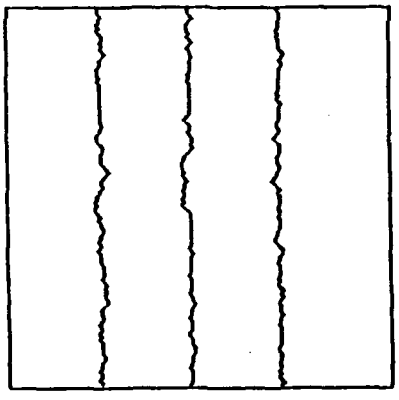


Initial state

Figure 4.2 The original image and the initial state.

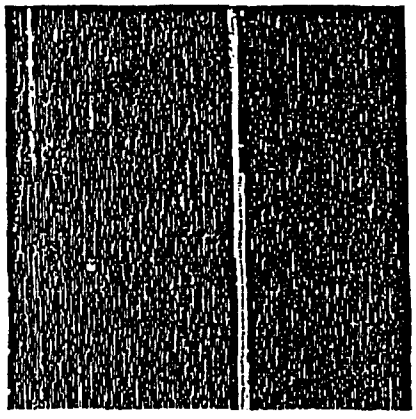


Final state for
N = 1800

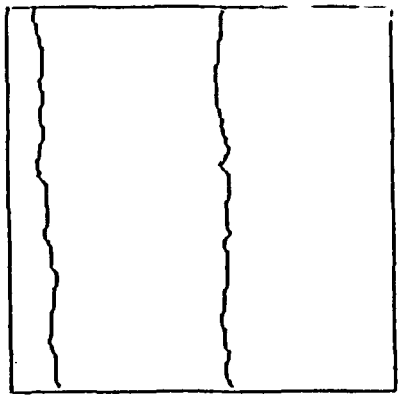


Final state for
9 repetitions/temp.
N = 200

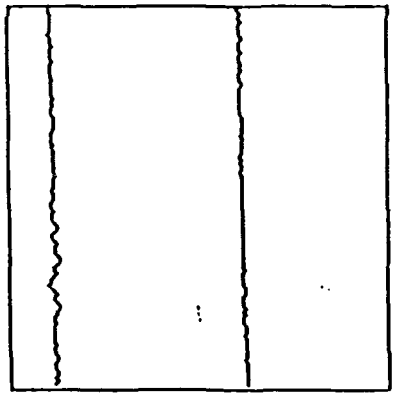
Figure 4.3 The results for three-track detection with SNR = -12 dB.



Two sweep tracks
with SNR = 0 dB .



Initial state



Final state for
9 repetitions/temp.
N = 200

Figure 4.4 The sweep-track detection.

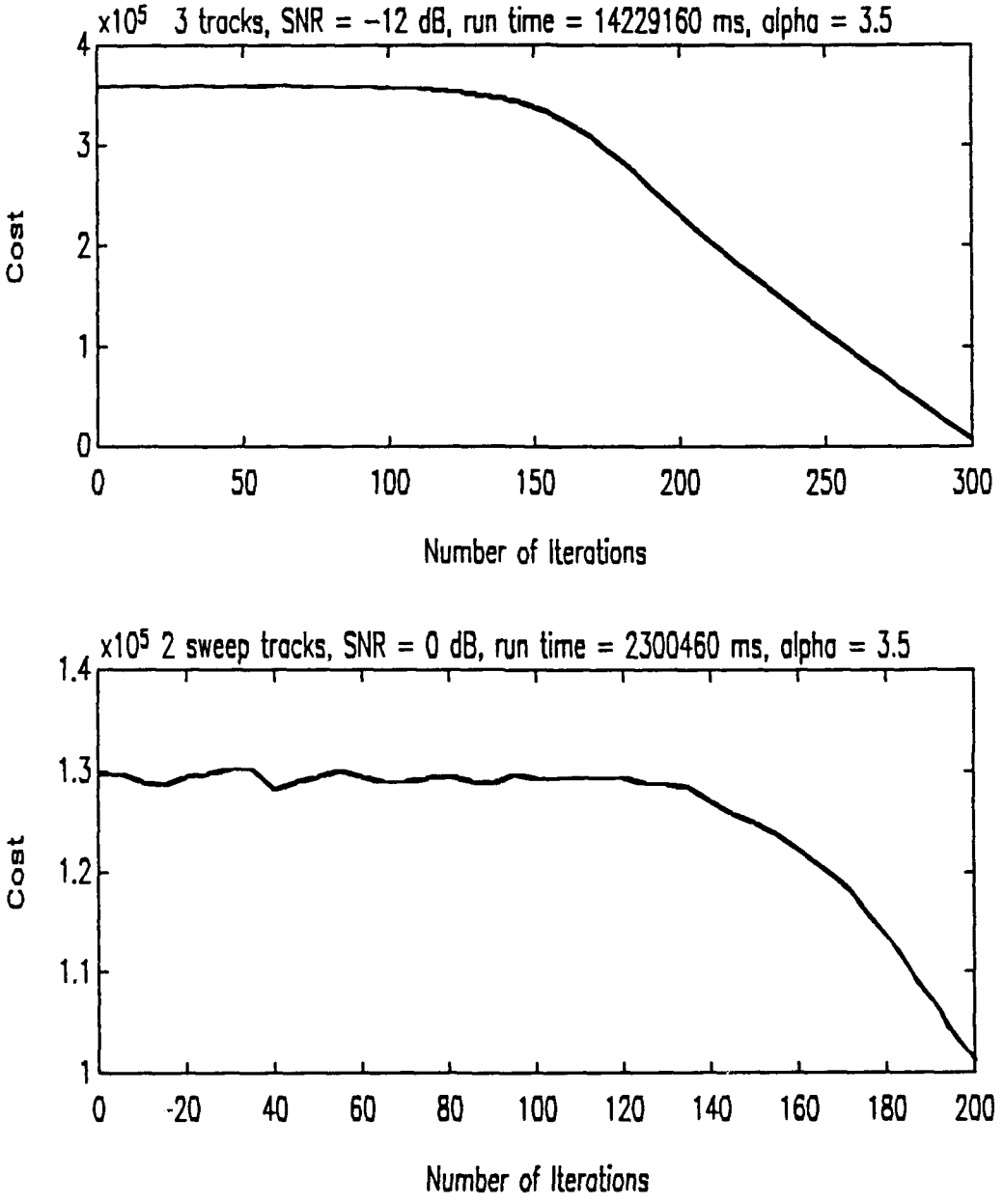
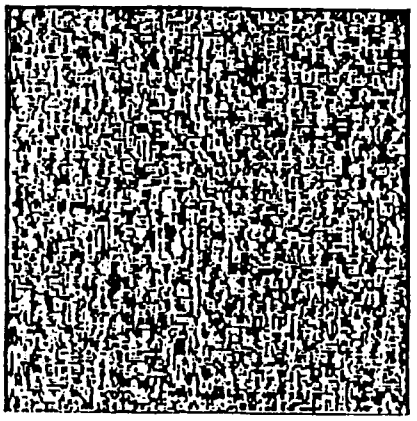
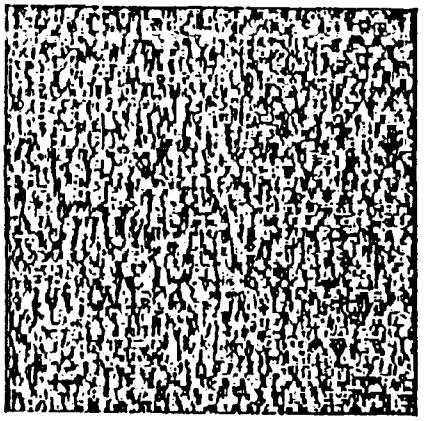


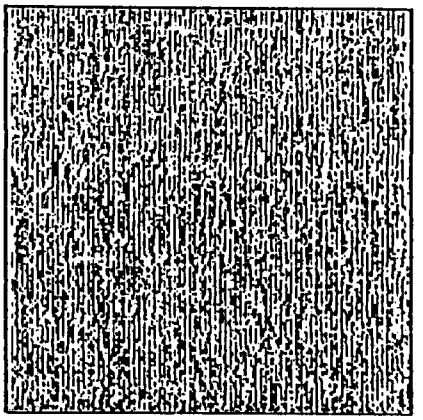
Figure 4.5 The cost variations for three-track data & sweep-track data.



Use Sobel
detector

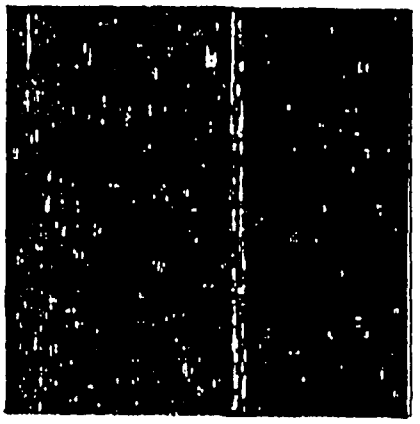


Use Robert
detector

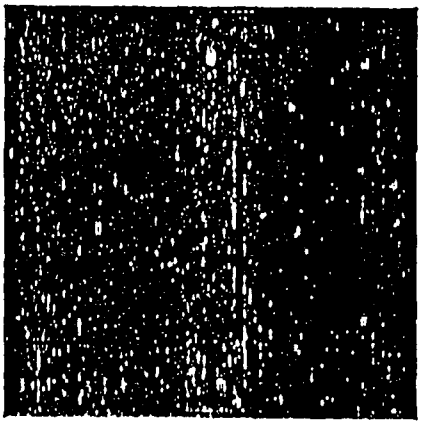


Use Laplacian
detector

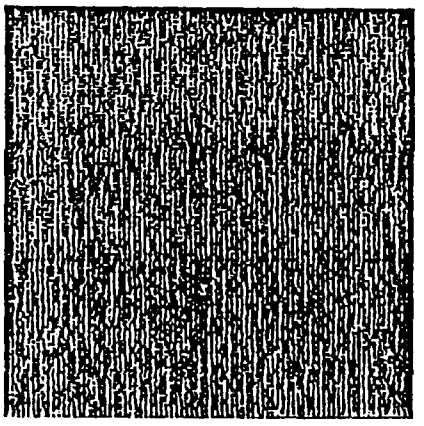
Figure 4.6 Three-track detection
by classical detectors.



Use Sobel
detector



Use Robert
detector



Use Laplacian
detector

Figure 4.7 Sweep-track detection by
classical edge detectors.

V CONCLUSIONS

A. Summary

The problem of sonar track detection in noise is difficult for classical edge detectors. This thesis uses a Simulated Annealing algorithm to solve the problem by formulating the sonar track detection problem as an optimization problem. Simulated Annealing is very effective in finding the global optimal solution, and the experiments conducted in this thesis showed fairly good results on data with very low SNR values(down to SNR = -18 dB). The experimental image data includes single track data, multi-track (3 tracks) data, and sweep-track data. The results for these data are fairly good. Lower values of SNR resulted in more deviation from the true track position. However, the results are still much better than those of classical edge detectors.

B. Recommendations

Some unsolved problems arose during the study, which need future work. They are:

1. The selection of the thresholding parameter α is currently a manual procedure. Some trial and error is necessary to get the best results. An automatic method is needed to choose an appropriate value of α effectively.
2. The cost variations for test data of very low SNR (below

-12 dB) cannot reach an asymptotic state even with long annealing times. However a fairly good result was still obtained. This is a very odd phenomena and needs more study to find a reasonable explanation.

3. Multi-track detection data with tracks crossing or touching each other cannot be handled in the algorithm implemented in this thesis. Some improvements to the algorithm need to be accomplished to overcome this limitation.
4. The use of Simulated Annealing in local track fine tuning in this thesis applies only to searching the track in a small region. A large region searching method using Simulated Annealing is needed to make the tracking algorithm more practical.

LIST OF REFERENCES

1. N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and D. Teller, "Equation of state calculation by fast computing machines," J. Chem. Phys. Vol. 21, pp. 1087 -1092(June 1953).

2. T. Hibbert, "Sonar Track Detection Using Simulated Annealing," Imperial College of Science and Technology, 1989.

3. S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi, "Optimization by Simulated Annealing," Science, Vol. 220, No. 4598, pp. 671-680 (13 May 1983).

4. Hin Leong Tan, Edward J. Delp, "Edge Detection by Cost Minimization," Purdue University, 1988.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Department Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5100	1
4.	Professor Chin-Hwa Lee, Code EC/Le Department of Electrical and Computer Engineering Monterey, CA 93943-5100	4
5.	Professor Charles W. Therrien, Code EC/Ti Department of Electrical and Computer Engineering Monterey, CA 93943-5100	1
6.	Chen, Tung-Sheng Department of Physics Chung Cheng Institute of Technology Ta-Hsi, Tao-Yuan, Taiwan, Republic of China	4
7.	Chung Cheng Institute of Technology, Library Ta-Hsi, Tao-Yuan, Taiwan, Republic of China	1